

**VŠB – Technická univerzita Ostrava**  
**Fakulta elektrotechniky a informatiky**

**BAKALÁŘSKÁ PRÁCE**

**2010**

**Jakub Měchura**

**VŠB – Technická univerzita Ostrava**  
**Fakulta elektrotechniky a informatiky**  
**Katedra informatiky**

**Modernizace a rozšíření internetových stránek**  
**Regionálního centra celoživotního vzdělávání**  
**RCCV Web Pages Modernization and Extension**

**2010**

**Jakub Měchura**

**Prohlášení**

Prohlašuji, že jsem tuto bakalářskou/diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

7. 5. 2010      Jakub Měchura

**Poděkování**

Na tomto místě bych rád poděkoval vedoucí své bakalářské práce paní Ing. Emílii Šeptákové za konzultace, pomoc a cenné rady při vytváření této práce.

## **Abstrakt**

Cílem této bakalářské práce je provést modernizaci webových stránek regionálního centra pro celoživotní vzdělávání nacházející se na adrese [www.rccv.vsb.cz](http://www.rccv.vsb.cz). Součástí modernizace je změna statických webových stránek na dynamicky generované stránky, jejichž součástí budou dva informační systémy.

První informační systém bude sloužit k vytváření kurzů a elektronickému podávání přihlášek na kurzy pořádané centrem pro celoživotní vzdělávání a jejich evidenci. V rámci systému správce vytváří kategorie, kurzy a termíny. Ty jsou dynamicky generovány na webových stránkách.

Druhý informační systém bude sloužit k evidování projektů, na kterém pracují zaměstnanci školy. V rámci systému bude možno vytvářet nové projekty, v rámci těchto projektů bude moci manager projektu vytvářet členy týmu pracující na jednotlivých projektech. Členové projektu budou moci vytvářet kategorie a ukládat dokumenty. Tyto projekty poté budou umístěny na jedné webové adrese.

## **Klíčová slova**

Databáze, datová analýza, lineární zápis, datový slovník, ER diagram, funkční analýza, kontextový diagram, data flow diagram, minispecifikace, časová analýza, SŘBD, indexová analýza, PHP, MySQL, Apache, XHTML, JavaScript, LDAP.

## **Abstract**

The aim of this bachelor thesis is to upgrade websites of Regional Lifelong Learning Center, which is located at <http://rccv.vsb.cz>. The part of modernization is a change from static websites in dynamically generated websites, whose parts will be two information systems.

The first information system will be useful for creating courses and electronic submitting an application on these courses. The system administrator creates categories, courses and terms. These items are dynamically generated on the website from database.

The second information system will be used to registration projects on which school employees is working. Within this system it will be possible to create new projects in which manager or administrator can create members of teams which will be working on these projects and members of teams can create categories and read and write some documents. These projects will be located at one web address.

## **Key words**

Databases, data analysis, linear registration, data dictionary, ER diagram, functional analysis, context diagram, data flow diagram, mini specification, temporal analysis, DBMS, index analysis, PHP, MySQL, Apache, XHTML, JavaScript, LDAP.

## **Seznam zkratk**

CSS	- Cascading Style Sheets, kaskádové styly
DFD	- Data Flow Diagram, diagram datových toků
E-R diagram	- Entity – Relationship diagram, diagram vztahů a entit
FK	- Foreign key, cizí klíč
HTML	- HyperText Markup Language, jazyk pro tvorbu hypertextů
IS	- Informační systém
LDAP	- Lightweight Directory Access Protocol, protokol pro ukládání a přístup k datům na adresářovém serveru
LAMP	- zkratka pro kombinaci technologií Linux, Apache, MySQL, PHP
Linux	- operační systém
MySQL	- databázový systém
PECL	- PHP Extension Community Library, knihovna funkcí
PHP	- HyperText Preprocessor, dynamický programovací jazyk
PK	- Primary key, primární klíč
RCCV	- Regionální centrum celoživotního vzdělávání
SHA1	- hashovací funkce
XHTML	- eXtensible HyperText Markup Language, jazyk pro tvorbu hypertextů

## Seznam obrázků

Obrázek 1 - IS přihlašování na kurzy - okolí systému .....	- 7 -
Obrázek 2 - IS přihlašování na kurzy - E-R diagram .....	- 8 -
Obrázek 3 - IS přihlašování na kurzy - kontextový diagram.....	- 13 -
Obrázek 4 - IS přihlašování na kurzy - DFD 0. úrovně.....	- 14 -
Obrázek 5 - IS přihlašování na kurzy - DFD 1. úrovně - správa kategorií.....	- 15 -
Obrázek 6 - IS přihlašování na kurzy - DFD 1. úrovně - správa kurzů.....	- 16 -
Obrázek 7 - IS přihlašování na kurzy - DFD 1. úrovně - správa termínů.....	- 17 -
Obrázek 8 - IS přihlašování na kurzy - DFD 1. úrovně - správa přihlášek .....	- 18 -
Obrázek 9 - Formulář pro vložení kategorie .....	- 19 -
Obrázek 10 – Formulář pro editování kategorie.....	- 19 -
Obrázek 11 – Formulář pro vložení kurzu a termínu .....	- 21 -
Obrázek 12 – Formulář pro editování kurzu či termínu .....	- 22 -
Obrázek 13 - Formulář pro odeslání přihlášky.....	- 24 -
Obrázek 14 - Seznam s přihláškami .....	- 24 -
Obrázek 15 - Seznam přihlášek uživatele .....	- 25 -
Obrázek 16 - Detail přihlášky .....	- 25 -
Obrázek 17 - Časová analýza stavu přihlášky.....	- 26 -
Obrázek 18 – Jak komunikují SŘBD .....	- 30 -
Obrázek 19 - IS přihlašování na kurzy - přihlášení do systému .....	- 33 -
Obrázek 20 - IS přihlašování na kurzy - úvodní obrazovka .....	- 34 -
Obrázek 21 - IS přihlašování na kurzy - sekce Users.....	- 34 -
Obrázek 22 - IS přihlašování na kurzy - sekce kategorie .....	- 35 -
Obrázek 23 – IS přihlašování na kurzy - kurzy a termíny tvorba.....	- 35 -
Obrázek 24 - IS přihlašování na kurzy - kurzy a termíny seznam .....	- 36 -
Obrázek 25 - IS přihlašování na kurzy - sekce přihláška .....	- 36 -
Obrázek 26 - IS přihlašování na kurzy - sekce přihláška - seznam přihlášek v kurzu .....	- 37 -
Obrázek 27 - IS přihlašování na kurzy – seznam lidí v kurzu pro tisk.....	- 37 -
Obrázek 28 - IS přihlašování na kurzy - změna hesla .....	- 37 -
Obrázek 29 - IS přihlašování na kurzy - odhlášení .....	- 38 -

Obrázek 30 - IS přihlašování na kurzy - uživatelské rozhraní .....	38 -
Obrázek 31 - IS přihlašování na kurzy - detail přihlášky .....	39 -
Obrázek 32 - WWW stránky – menu .....	39 -
Obrázek 33 WWW stránky – kurzy pro veřejnost .....	40 -
Obrázek 34 - WWW stránky - seznam kurzů v rámci kategorie.....	40 -
Obrázek 35 - WWW stránky - detail o Kurzu .....	40 -
Obrázek 36 - WWW stránky - přihláška na kurz .....	41 -
Obrázek 37 – IS projekty - okolí systému.....	47 -
Obrázek 38 - IS projekty - E-R diagram .....	48 -
Obrázek 39 - IS projekty - kontextový diagram.....	53 -
Obrázek 40 - IS projekty - DFD 0. Úrovně.....	54 -
Obrázek 41 - IS projekty - DFD 1. úrovně - správa uživatelů.....	55 -
Obrázek 42 - IS projekty - DFD 1. úrovně - správa projektů.....	56 -
Obrázek 43 - IS projekty – DFD 1. úrovně – správa kategorií.....	57 -
Obrázek 44 - IS projekty - DFD 1. úrovně správa dokumentů.....	58 -
Obrázek 45 - IS projekty - DFD 1. úrovně – příspěvky .....	59 -
Obrázek 46 - IS projekty - obecné řešení přístupu k funkcím.....	60 -
Obrázek 47 - IS projekty - formulář pro vkládání projektu.....	60 -
Obrázek 48 - IS projekty - seznam projektů.....	61 -
Obrázek 49 - IS projekty - formulář pro editování projektu.....	62 -
Obrázek 50 - IS projekty - seznam projektů.....	63 -
Obrázek 51 - IS projekty - formulář pro vložení člena týmu .....	63 -
Obrázek 52 - IS projekty - formulář pro odesílání příspěvku.....	64 -
Obrázek 53 - IS projekty - seznam přečtené pošty.....	65 -
Obrázek 54 - Časová analýza stavu projektu .....	65 -
Obrázek 55 - LDAP příklad .....	67 -
Obrázek 56 - IS projekty - přihlašování do systému .....	71 -
Obrázek 57 - IS projekty - úvodní obrazovka .....	71 -
Obrázek 58 - IS projekty - sekce projekty.....	72 -
Obrázek 59 - IS projekty - sekce uživatelé.....	73 -



Obrázek 60 - IS projekty - sekce kategorie před vybráním projektu.....	- 73 -
Obrázek 61 - IS projekty - sekce kategorie po vybrání projektu.....	- 74 -
Obrázek 62 - IS projekty - seznam kategorií.....	- 74 -
Obrázek 63- IS projekty - sekce dokumenty .....	- 74 -
Obrázek 64 - IS projekty - sekce pošta.....	- 75 -
Obrázek 65 - IS projekty - sekce nepřečtená pošta .....	- 75 -
Obrázek 66 - IS projekty - detail pošty .....	- 75 -
Obrázek 67 - IS projekty - sekce osobní .....	- 76 -

## Obsah

<b>1</b>	<b>Úvod .....</b>	<b>- 2 -</b>
<b>2</b>	<b>Informační systém pro jednoduchou evidenci přihlášek na kurzy .....</b>	<b>- 4 -</b>
<b>2.1</b>	<b>Zadání.....</b>	<b>- 4 -</b>
2.1.1	Proč? .....	- 4 -
2.1.2	K čemu?.....	- 4 -
2.1.3	Kdo? .....	- 4 -
2.1.4	Vstupy do systému .....	- 4 -
2.1.5	Výstupy ze systému.....	- 5 -
2.1.6	Vybrané funkce systému .....	- 6 -
2.1.7	Okolí systému.....	- 7 -
2.1.8	Nefunkční požadavky .....	- 7 -
<b>2.2</b>	<b>Datová analýza.....</b>	<b>- 8 -</b>
2.2.1	E-R diagram .....	- 8 -
2.2.2	Lineární zápis typů entit .....	- 9 -
2.2.3	Lineární zápis typů vztahů.....	- 9 -
2.2.4	Datový slovník .....	- 9 -
<b>2.3</b>	<b>Funkční analýza.....</b>	<b>- 13 -</b>
2.3.1	Kontextový diagram .....	- 13 -
2.3.2	DFD .....	- 13 -
2.3.3	Minispecifikace .....	- 19 -
2.3.4	Časová analýza stavu přihlášek .....	- 26 -
<b>2.4</b>	<b>Návrh implementace .....</b>	<b>- 27 -</b>
2.4.1	Použité technologie .....	- 27 -
2.4.2	Jak komunikují SŘBD.....	- 30 -
2.4.3	Hardware, Software .....	- 30 -
2.4.4	Změna v datových slovnících.....	- 31 -
2.4.5	Indexová analýza .....	- 31 -
<b>2.5</b>	<b>Popis implementace .....</b>	<b>- 33 -</b>

2.5.1	Uživatelská příručka.....	- 33 -
2.5.2	Programátorská příručka .....	- 41 -
2.5.3	Popis instalace .....	- 41 -
<b>3</b>	<b>Informační systém pro evidenci projektů.....</b>	<b>- 44 -</b>
<b>3.1</b>	<b>Zadání.....</b>	<b>- 44 -</b>
3.1.1	Proč?.....	- 44 -
3.1.2	K čemu?.....	- 44 -
3.1.3	Kdo? .....	- 44 -
3.1.4	Vstupy .....	- 44 -
3.1.5	Výstupy .....	- 45 -
3.1.6	Vybrané funkce systému .....	- 46 -
3.1.7	Okolí systému.....	- 47 -
<b>3.2</b>	<b>Datová analýza.....</b>	<b>- 48 -</b>
3.2.1	E-R diagram .....	- 48 -
3.2.2	Lineární zápis typů entit .....	- 49 -
3.2.3	Lineární zápis typů vztahů.....	- 49 -
3.2.4	Datový slovník .....	- 50 -
<b>3.3</b>	<b>Funkční analýza.....</b>	<b>- 53 -</b>
3.3.1	Kontextový diagram .....	- 53 -
3.3.2	DFD .....	- 54 -
3.3.3	Vybrané minispecifikace .....	- 60 -
3.3.4	Časová analýza stavu projektu .....	- 65 -
<b>3.4</b>	<b>Návrh implementace .....</b>	<b>- 66 -</b>
3.4.1	Použité technologie .....	- 66 -
3.4.2	Hardware, Software.....	- 68 -
3.4.3	Změna v datových slovnících.....	- 69 -
3.4.4	Indexová analýza .....	- 69 -
<b>3.5</b>	<b>Popis Implementace .....</b>	<b>- 71 -</b>
3.5.1	Uživatelská příručka.....	- 71 -
3.5.2	Programátorská příručka .....	- 76 -

3.5.3	Popis instalace .....	- 76 -
<b>4</b>	<b>Závěr .....</b>	<b>- 78 -</b>
<b>5</b>	<b>Zdroje a přílohy .....</b>	<b>- 80 -</b>
5.1	Literatura .....	- 80 -
5.2	Internet .....	- 80 -
5.3	Seznam příloh na CD .....	- 81 -



# 1 Úvod

Tato bakalářská práce se zabývá modernizací webových stránek, respektive změnou statických webových stránek na dynamicky generované, jejíž součástí budou dva informační systémy.

První informační systém je vyvíjen pro zlepšení systému přihlašování na kurzy „Centra pro celoživotní vzdělávání“, vytváření kategorií, kurzů a termínů v rámci kurzů, elektronického podávání přihlášek na kurzy a evidenci těchto kurzů a přihlášek. Vytvořené kategorie, kurzy a termíny jsou automaticky generovány na webových stránkách, což značně ulehčí práci s ručním vytvářením webových stránek a počtem takto vytvořených stránek.

Druhý informační systém bude sloužit k evidování projektů, na kterém pracují zaměstnanci školy. V rámci systému bude možno vytvářet nové projekty, v rámci těchto projektů potom členy týmu, kategorie a vkládat dokumenty. Výhodou bude umístění informací o všech projektech na jedné webové adrese.

Následující kapitola se zabývá tvorbou prvního informačního systému. Skládá z několika podkapitol, a to zadání, v nichž je specifikováno zadání tohoto IS, k čemu by měl sloužit a co by mělo být jeho výsledkem. Dále pak obsahuje datovou analýzu zabývající se vývojem databáze na logické úrovni, tj. definici tabulek, jejich atributů a vztahů mezi nimi. Další podkapitolou je funkční analýza, která se zabývá popisem akcí, provádějící se nad danou databází. Předposlední podkapitolou je návrh implementace, v nichž jsou popsány technologie použité pro vývoj IS, dále pak použitý software a hardware, změny v datových slovnících a indexovou analýzu. Poslední podkapitolou je popis implementace, který obsahuje uživatelskou a programátorskou příručku, popis instalace a nasazení IS do provozu.

Třetí kapitola se zabývá návrhem informačního systému evidence projektů a má stejnou strukturu jako kapitola předcházející.

V kapitole závěr se zabývám shrnutím a zhodnocením celého vývoje obou informačních systémů.

Na konci práce je seznam použité literatury, seznam příloh na přiloženém CD a obsah těchto příloh.



## 2 Informační systém pro jednoduchou evidenci přihlášek na kurzy

### 2.1 Zadání

V této kapitole je specifikováno proč a k čemu je daný systém vyvíjen, kdo s ním bude pracovat a jaké jsou požadavky na něj, a to jak po stránce funkční, tak po stránce provozní.

#### 2.1.1 Proč?

Momentálně je přihlašování na kurzy pořádané RCCV při VŠB – TUO prováděno formou předchystaného formuláře umístěného na stránkách [www.rccv.vsb.cz](http://www.rccv.vsb.cz). Každý, kdo se chce se přihlásit na nějaký kurz, musí si vytisknout a vyplnit předchystaný formulář a odevzdat, popř. zaslat na příslušné místo. To má za následek velké nároky na uskladnění, přehlednost a vyhledávání. Proto je potřeba vytvořit IS pro elektronické přihlašování veřejnosti na jednotlivé kurzy.

#### 2.1.2 K čemu?

K rychlejšímu zpracování, lepší přehlednosti přihlášek uživatelů, jejich lepší skladnosti a rychlejšímu dohledávání v případě nejasností. K usnadnění práce a správy webových stránek.

#### 2.1.3 Kdo?

S IS budou pracovat uživatelé v následujících rolích:

- **Návštěvník** – přihlašuje se na jednotlivé kurzy pořádané RCCV
- **Účastník kurzů** – může se přihlašovat do systému, prohlížet si a kontrolovat stav přihlášek
- **Správce** – člověk starající se o IS

#### 2.1.4 Vstupy do systému

U kategorie budeme evidovat jednoznačné číslo kategorie a její název.

U kurzu budeme evidovat jednoznačné číslo kurzu, název kurzu, popis kurzu, jeho délku, počet hodin kurzu, jméno garanta, cenu, jednoznačné číslo kategorie, formu kurzu, pro koho je kurz určen, formu ukončení kurzu, předběžné požadavky na kurz, obsah studijního balíku, jak bude kurz probíhat a „zisky (přínosy)“ kurzu.

U termínu evidujeme jednoznačné číslo termínu, datum zahájení, popř. jiný termín zahájení, místo konání termínu, kapacitu kurzu v daném termínu a jednoznačné číslo kurzu.

U lidí budeme evidovat jednoznačné číslo uživatele (správce), jeho jméno, příjmení, heslo, ulici, město, PSČ, telefon, email, firmu, kde uživatel pracuje a jednoznačné číslo role.

U přihlášky budeme evidovat jednoznačné číslo přihlášky, jednoznačné číslo kurzu, jednoznačné číslo termínu, jednoznačné číslo lidí, způsob úhrady, stav přihlášky, zda je přihláška zaplacená, datum podání přihlášky a poznámku.

U role budeme evidovat jednoznačné číslo role a druh role.



V kategorii může být vedeno více kurzů, ovšem stejný kurz nemůže být ve více kategoriích.

Kurz může být pořádán ve více termínech.

Na jeden termín může být podáno n přihlášek. Přihláška se však vztahuje jen k jednomu termínu.

Lidé mohou podat více přihlášek, jedna přihláška je ale jen pro jednoho člověka.

Každý člověk má jednu roli.

### 2.1.5 Výstupy ze systému

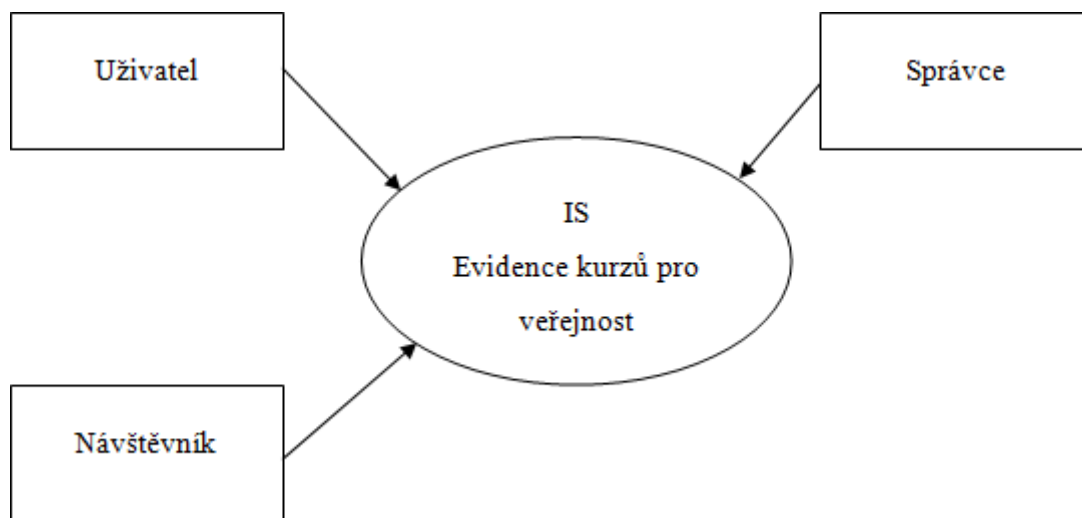
- **Seznam kategorií** – v menu budou zobrazeny všechny aktuální kategorie.
- **Seznam kurzů** – v rámci kategorie bude zobrazen seznam kurzů.
- **Informace o kurzech** – po kliknutí na každý kurz se zobrazí uživateli podrobné informace o kurzu jako jsou popis kurzu, jeho garant, délka, cena aj.
- **Seznam termínu a informace o něm** – v rámci každého kurzu bude seznam aktuálních termínů, u každého termínu budou informace o tom, kdy a kde se bude kurz konat a jaká je jeho kapacita.
- **Seznam přihlášek** – při přihlášení do IS bude mít uživatel k dispozici seznam svých doposud podaných přihlášek, po kliknutí na jednotlivé přihlášky se zobrazí jejich detail.
- **Seznam studentů kurzu v daném termínu** – seznam všech studentů v daném kurzu
- **Seznam všech uživatelů, kategorií, kurzů a termínů, přihlášek** – správce bude mít k dispozici seznam všech aktuálních kategorií, kurzů a termínů, uživatelů a jejich přihlášek.

### 2.1.6 Vybrané funkce systému

Událost	Reakce	Aktér
Nový účastník kurzu	Zapiš do seznamu přihlášek, lidí	Návštěvník
Smazání účastníka kurzu	Smaž přihlášku, účastníka kurzu ze seznamu přihlášek a uživatelů	Správce
Výpis lidí	Zobraz seznam uživatelů	Správce
Smazání přihlášky	Smaž přihlášku ze seznamu přihlášek	Správce
Editace přihlášky	Edituj danou přihlášku	Správce
Výpis prezentační listiny	Zobraz seznam přihlášených na daný kurz v daném termínu	Správce
Výpis přihlášek	Zobraz uživateli přihlášky	Účastník kurzu, správce
Stav přihlášky	Změň stav přihlášky v seznamu přihlášek (přijata, zpracována apod.)	Správce
Zaplacení přihlášky	Nastav atribut zaplacení v tabulce Přihláška na true	Správce
Tisk přihlášek	Tiskni přihlášku	Účastník kurzu, správce
Smazání kategorie	Smaž kategorii ze seznamu kategorií	Správce
Editace kategorie	Edituj danou kategorii	Správce
Výpis kategorií	Zobraz kategorie	Správce, Návštěvník
Editace kurzu	Edituj daný kurz	Správce
Výpis kurzu	Zobraz kurzy pro dané kategorie	Správce, Návštěvník
Smazání termínu	Smaž termín ze seznamu termínů	Správce
Nový termín	Přidej termín do seznamu termínů	Správce
Editace termínu	Edituj daný termín	Správce

Výpis termínu	Zobraz termín pro danou kategorii a termín	Správce, Návštěvník
---------------	--	---------------------

### 2.1.7 Okolí systému



Obrázek 1 - IS přihlašování na kurzy - okolí systému

### 2.1.8 Nefunkční požadavky

Výsledný IS by měl být podle nejnovějších standardů a co nejlevnější. Předpokládá se přístup více uživatelů – klient server implementace, přístup přes internet, rychlá odezva, příjemné a jednoduché ovládání a uživatelské prostředí.

Celý informační systém bude vyvíjen pomocí jazyku PHP a MySQL databáze, viz. kapitola

#### 2.4.1 Použité technologie

## 2.2 Datová analýza

Kapitola datová analýza se zabývá procesem poznávání objektů reálného světa, jejich vlastností a vazeb pomocí tabulek, atributu a propojení, která představuje nějakou strukturu databáze.

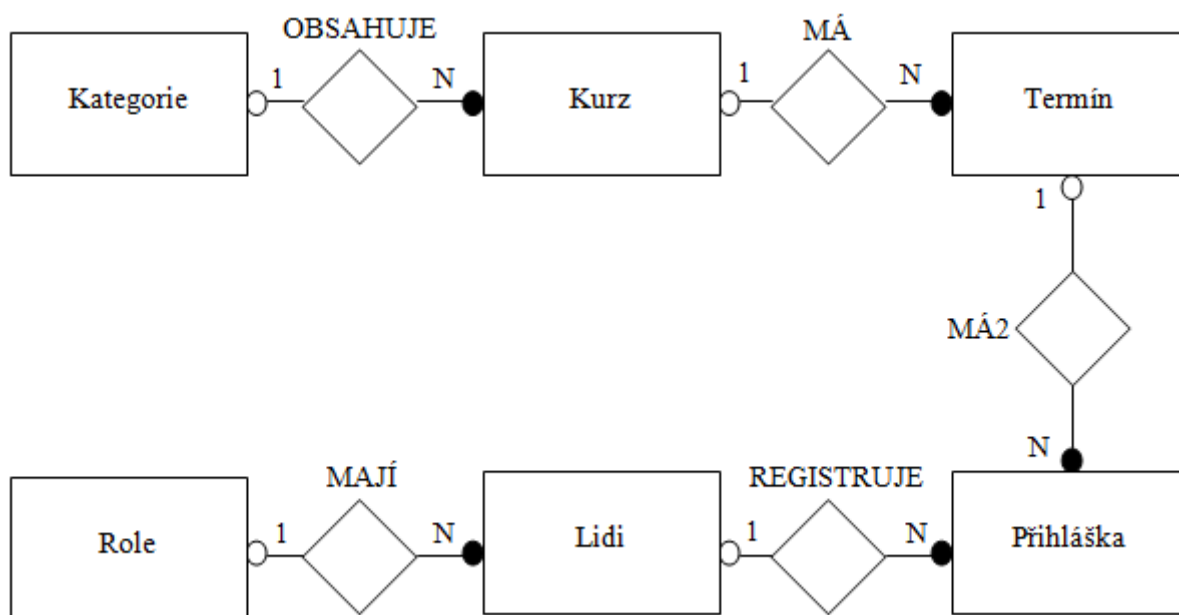
### 2.2.1 E-R diagram

Často používaný model pro popis logické struktury dat na konceptuální úrovni. Popisuje objekty a jejich vztahy buď textovým zápisem nebo pomocí E-R diagramů.

**E-R diagram se skládá z:**

- typu entity, který má tvar obdélníkového uzlu
- typu vztahu mezi entitami ve tvaru kosočtvercového uzlu
- kardinality vztahů, např. 1:1 nebo 1:N
- povinnost členství ve vztahu, které může být povinné či nepovinné.

Více podrobnosti o tvorbě datové analýzy lze nalézt v [11, str. 34][10, str. 25].



Obrázek 2 - IS přihlašování na kurzy - E-R diagram

### 2.2.2 Lineární zápis typů entit

Popisuje entity a vztahy mezi entitami.

#### Primární klíč, cizí klíč

Kategorie (**id\_kategorie**, název\_kategorie)

Kurz (**id\_kurzu**, název\_kurzu, popis, délka, počet\_hodin, garant, cena, **id\_kategorie**, forma\_kurzu, určeno\_pro, forma\_ukončení, požadavky, obsah\_balíku, průběh\_kurzu, zisky\_kurzu)

Termín (**id\_termínu**, zahájení\_termínu, kapacita, místo, jiný\_termín, **id\_kurzu**)

Přihláška (**id\_přihlášky**, **id\_kurzu**, **id\_termínu**, **id\_lidi**, způsob\_úhrady, stav, zaplaceno, datum\_podání, poznámka)

Lidi (**id\_lidi**, jméno, příjmení, heslo, ulice, město, PSČ, telefon, email, název\_firmy, **id\_role**)

Role (**id\_role**, role)

### 2.2.3 Lineární zápis typů vztahů

OBSAHUJE (Kategorie, Kurz)

MÁ (Kurz, Termín)

MÁ2 (Termín, Přihláška)

MÁ3(Kurz, Hodnoticí\_formulář)

REGISTRUJE (Přihláška, Lidi)

MAJÍ (Lidi, Role)

### 2.2.4 Datový slovník

Soubor, který definuje strukturu a složení datové základny a obsahuje meta data potřebná pro správu dat. Datový slovník zahrnuje seznam všech datových objektů v databázi, jména a popis všech datových prvků a jejich vztahů, údaje o integritních omezeních atd.

#### Kategorie – seznam kategorií obsahující kurzy

název	typ	vel.	klíč	null	index	popis
id_kategorie	int	5	PK	N	A	jednoznačná identifikace kategorie
název_kategorie	varchar	40	N	N	A	název uživatele

**Role – role uživatelů používající IS**

název	typ	vel.	klíč	null	index	popis
id_role	int	5	PK	N	A	jednoznačná identifikace role
název_role	varchar	15	N	N	A	název role

**Kurz – seznam kurzu v daná kategorii**

název	typ	vel.	klíč	null	index	popis
id_kurzu	int	10	PK	N	A	jednoznačná identifikace kurzu
název_kurzu	varchar	50	N	N	A	název kurzu
popis	varchar	1000	N	N	N	popis kurzu
délka	varchar	50	N	N	N	délka kurzu
počet_hodin	int	10	N	N	N	počet hodin daného kurzu v daném termínu
garant	varchar	50	N	N	N	garant kurzu
cena	varchar	10	N	N	N	cena kurzu
id_kategorie	int	5	FK	N	A	jednoznačná identifikace kategorie, cizí klíč z tabulky Kategorie
forma_kurzu	varchar	20	N	N	N	forma kurzu – prezenční, dálková apod.
určeno_pro	varchar	1000	N	N	N	pro koho je kurz určen
forma_ukončení	varchar	1000	N	N	N	forma ukončení kurzu
požadavky	varchar	1000	N	N	N	požadavky (předpoklady) pro zvládnutí kurzu
obsah_balíku	varchar	1000	N	N	N	obsah studijního balíku kurzu
průběh_kurzu	varchar	1000	N	N	N	průběh kurzu
zisky_kurzu	varchar	1000	N	N	N	co získá účastník kurzu jeho absolvováním

**Lidi - seznam lidí, kteří podali přihlášku na nějaký kurz**

<b>název</b>	<b>typ</b>	<b>vel.</b>	<b>klíč</b>	<b>null</b>	<b>index</b>	<b>Popis</b>
id_lidi	int	5	PK	N	A	jednoznačná identifikace Lidi
jméno	varchar	20	N	N	A	jméno přihlášeného
příjmení	varchar	20	N	N	A	příjmení přihlášeného
heslo	char	40	N	N	N	heslo pro přihlášení do IS
ulice	varchar	50	N	N	N	adresa přihlášeného – ulice
město	varchar	30	N	N	N	adresa přihlášeného – město
PSC	int	5	N	N	N	adresa přihlášeného – PSC
telefon	varchar	13	N	N	N	telefon ve tvaru +429999999999
email	varchar	40	N	N	N	email přihlášeného
název_firmy	varchar	200	N	A	N	název firmy přihlašovaného
id_role	int	5	FK	N	A	jednoznačná identifikace role, cizí klíč z tabulky Role

**Termín – seznam termínu, ve kterém daný termín probíhá**

<b>název</b>	<b>typ</b>	<b>vel.</b>	<b>klíč</b>	<b>null</b>	<b>index</b>	<b>popis</b>
id_termínu	int	5	PK	N	A	jednoznačná identifikace termínu
zaháje- ní_termínu	date		N	N	N	datum zahájení termínu
kapacita	int	10	N	N	N	kapacita kurzu v daném termínu
místo	varchar	100	N	N	N	místo kurzu v daném termínu
jiný_termín	varchar	30	N	N	N	jiný termín, např. kdykoliv
id_kurzu	int	5	FK	N	A	jednoznačná identifikace kurzu, cizí klíč z tabulky Kurz

cena	int	10	N	N	N	cena kurzu v daném termínu
------	-----	----	---	---	---	----------------------------

**Přihláška – přihláška podaná návštěvníkem na nějaký kurz**

název	typ	vel.	klíč	null	index	popis
id_přihlášky	int	5	PK	N	A	jednoznačná identifikace přihlášky
id_kurzu	int	5	FK	N	A	jednoznačná identifikace kurzu, cizí klíč z tabulky Kurz
id_termínu	int	5	FK	N	A	jednoznačná identifikace termínu, cizí klíč z tabulky Termín
id_lidi	int	5	FK	N	A	jednoznační identifikace lidi, cizí klíč z tabulky Lidi
způsob_úhrady	varchar	15	N	N	N	způsob úhrady přihlášeného – osobně, poštou, na účet
stav	varchar	15	N	N	N	stav přihlášky – přijatá, zpracovaná atd.
zaplacen	bool		N	N	N	stav zaplacení přihlášky
datum_podání	date		N	N	N	datum podání přihlášky
poznámka	varchar	255	N	A	N	poznámka k přihlášce



## 2.3 Funkční analýza

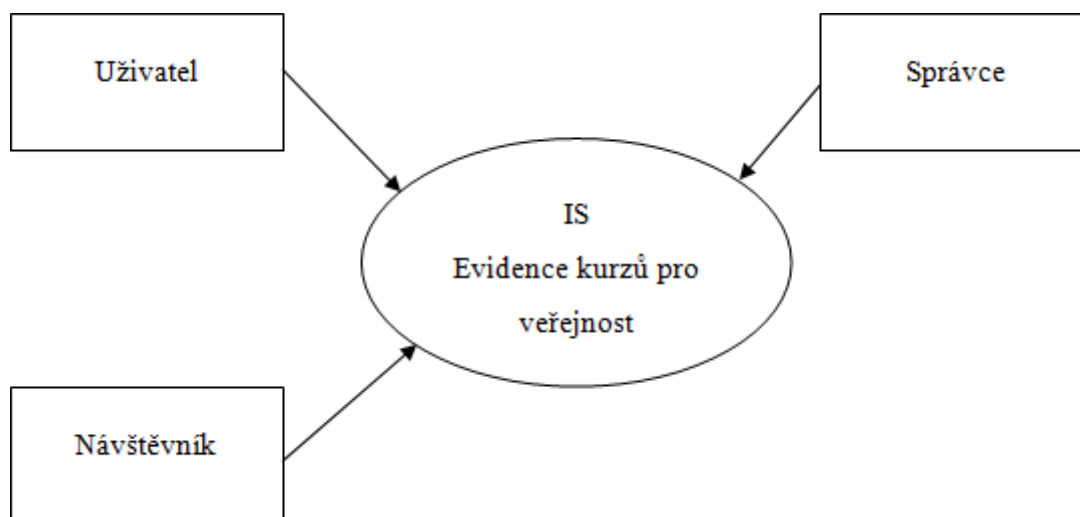
Představuje chování objektů v reálném světě. Popisuje akce, které jsou prováděny nad objekty reálného světa. Výsledkem je pojmenování a popis akcí, provádějící se nad danou databází[10, str. 27].

### 2.3.1 Kontextový diagram

Standardním nástrojem pro grafické zobrazení systému a jeho okolí je **kontextový diagram**.

**Základní prvky kontextového diagramu jsou:**

- obdélníkový uzel zobrazuje aktéra, představující zdroj nebo cíl informací pro IS
- kruhový uzel, který zobrazuje celý systém jako proces
- orientované hrany znázorňující datové toky



Obrázek 3 - IS přihlašování na kurzy - kontextový diagram

### 2.3.2 DFD

První vnější pohled na funkce se vytváří pomocí DFD. Jedná se o grafický prostředek pro návrh a zobrazení funkčního modelu systému. DFD by měl být dostatečně jednoduchý a názorný i pro uživatele. Může sloužit i k upřesnění zadání.

**DFD se skládá z následujících základních prvků:**

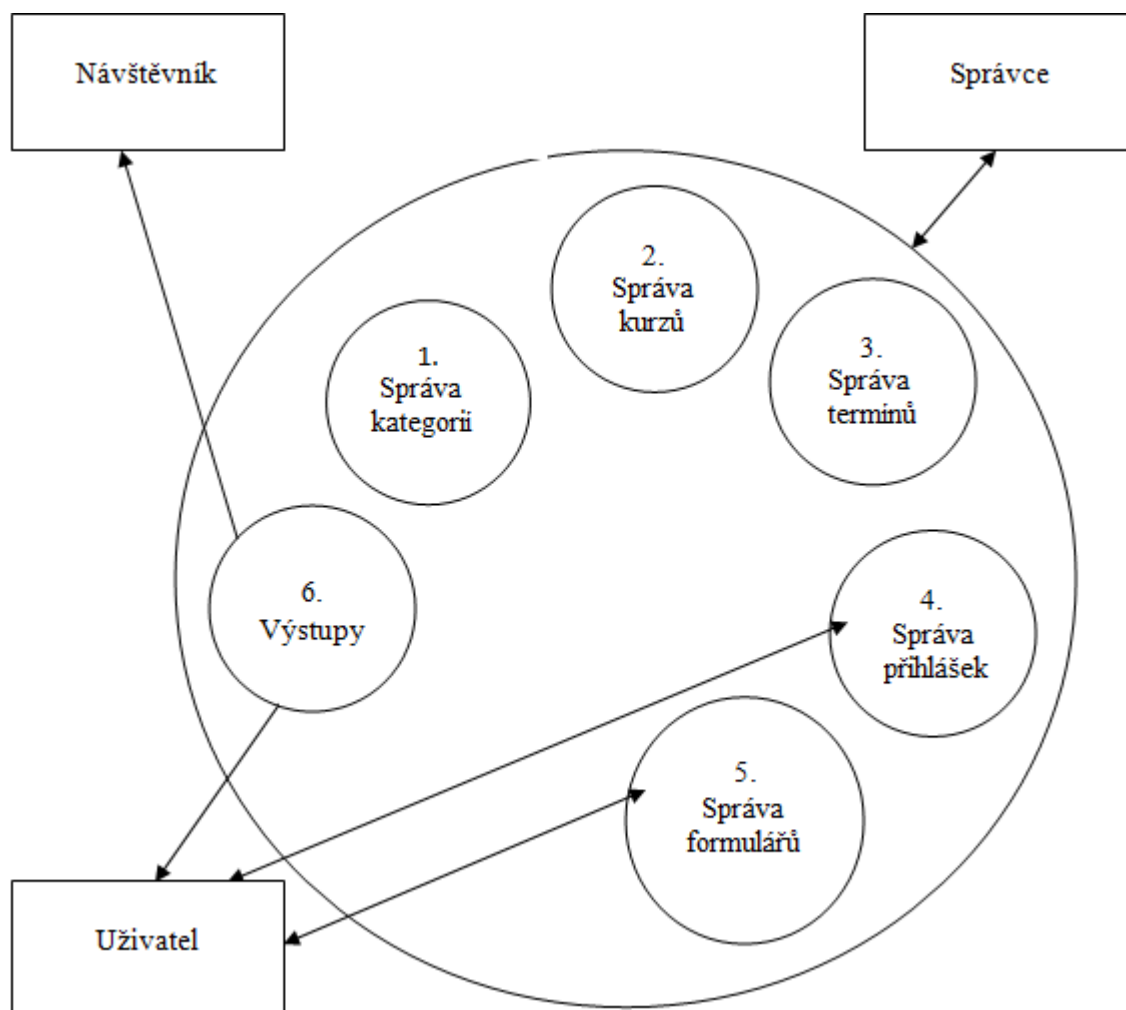
- **Funkce** - provádí transformaci dat vstupních na data výstupní, realizuje nějakou funkci nad daty. Zakresluje se kruhovým uzlem v grafu (elipsou), v uzlu je název funkce.
- **Datový tok** - vyjadřuje přesun dat z jedné části systému do jiné, z okolí systému do systému nebo ze systému do jeho okolí. Bývá zakreslen hranou (úsečkou) se šipkou, znamenající směr toku dat. Jestliže tečou data oboustranně, použijí se šipky s obou stran.
- **Datová paměť** - místo dočasného nebo trvalého uložení dat pro jejich pozdější použití. Může být implementován jako pole, textový soubor atd. Využívá se tehdy, kdy dochází ke zpoždění

při předávání dat. Znázorňuje se pomocí dvou rovnoběžek. Mezi nimi je umístěn název paměti.

- **Aktér** – jedná se o objekt, který se systémem komunikuje, např. člověk, jiný systém apod.

Více podrobnosti o tvorbě funkční analýzy lze nalézt zde [10, str. 28].

#### 2.3.2.1 DFD 0. úrovně

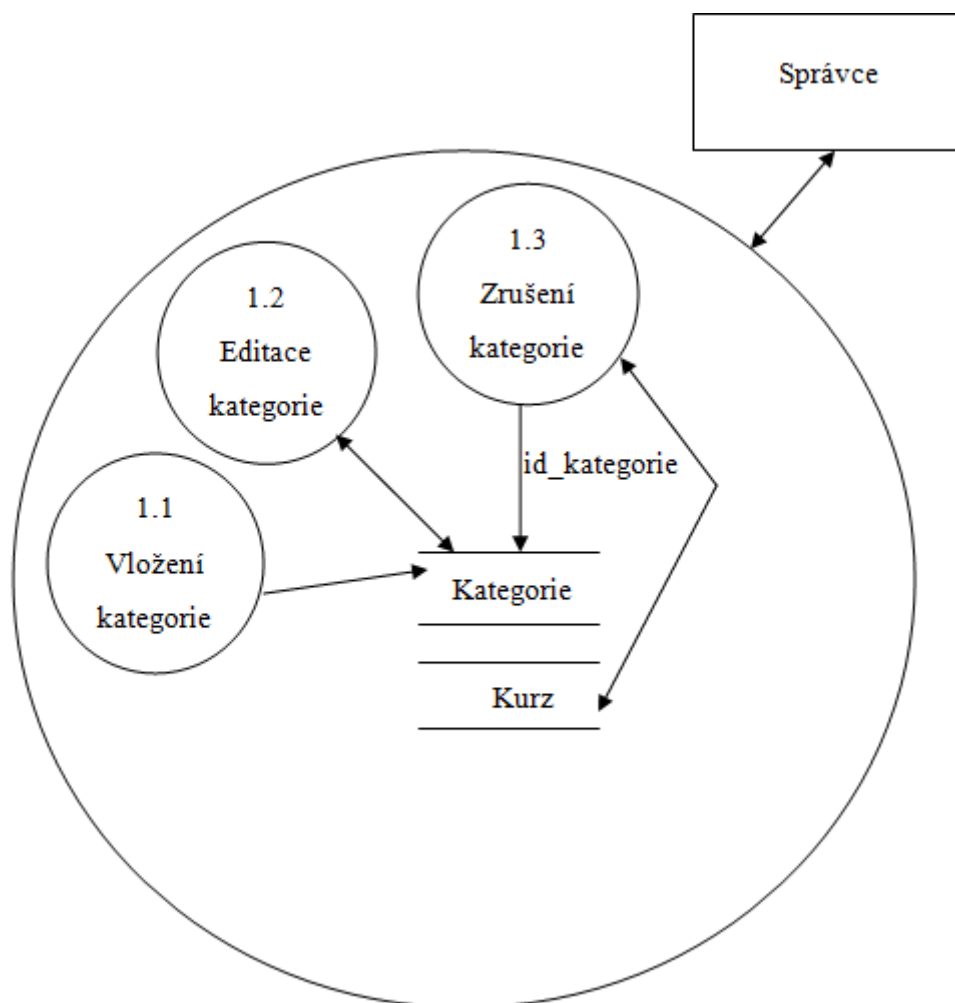


**Obrázek 4 - IS přihlašování na kurzy - DFD 0. úrovně**

V DFD 0. úrovně je možno vidět rozdělení IS do modulů jako správa kategorií, kurzů apod. Ty pak detailněji popisují tok dat nad daným modulem. Dále pak je možno vidět, kdo s jakým modulem smí pracovat.

### 2.3.2.2 DFD 1. úrovně

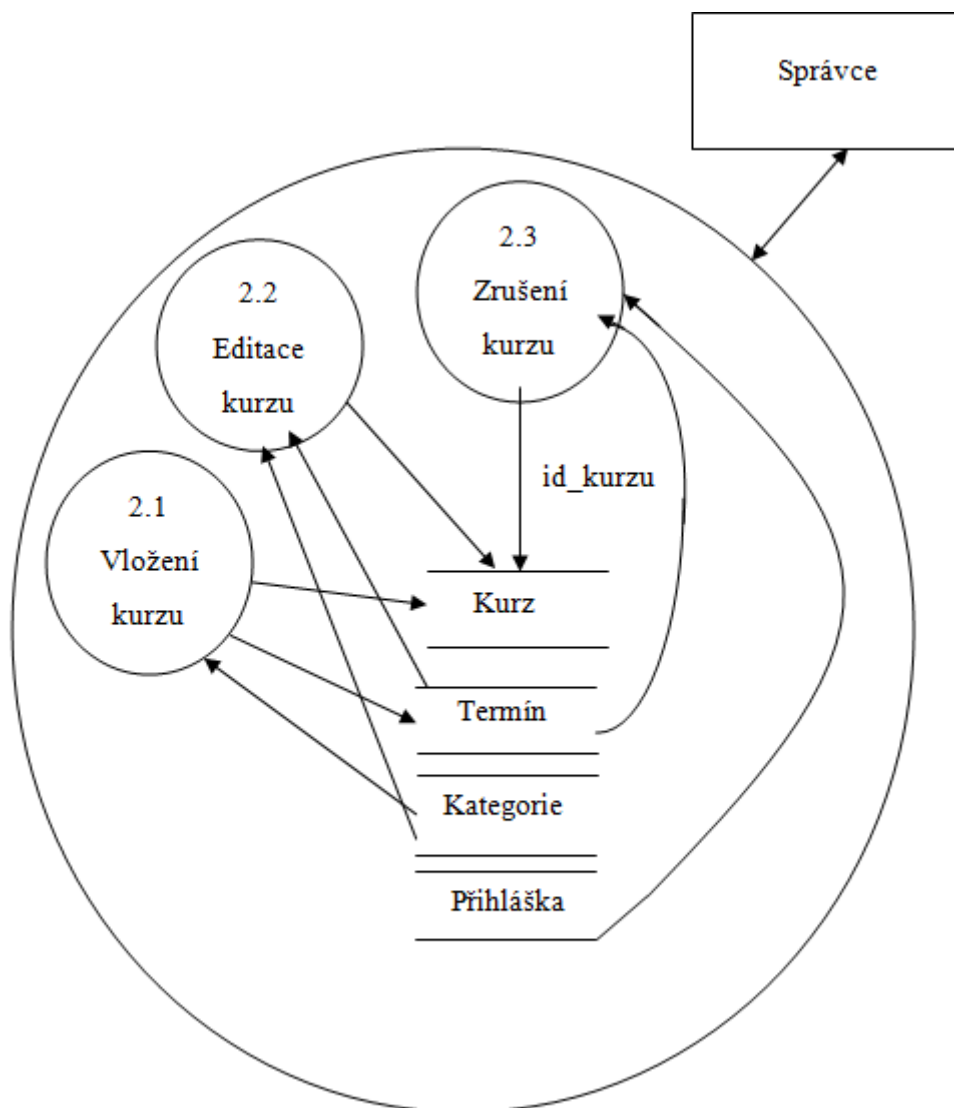
#### Správa kategorií



Obrázek 5 - IS přihlašování na kurzy - DFD 1. úrovně - správa kategorií

DFD 1. Úrovně – Správa kategorií se zabývá akcemi vkládáním, editací a zrušením kategorií. Zrušení kategorie nebude možné, jestliže bude obsahovat již nějaký kurz. Všechny tyto akce bude moci provádět pouze správce systému.

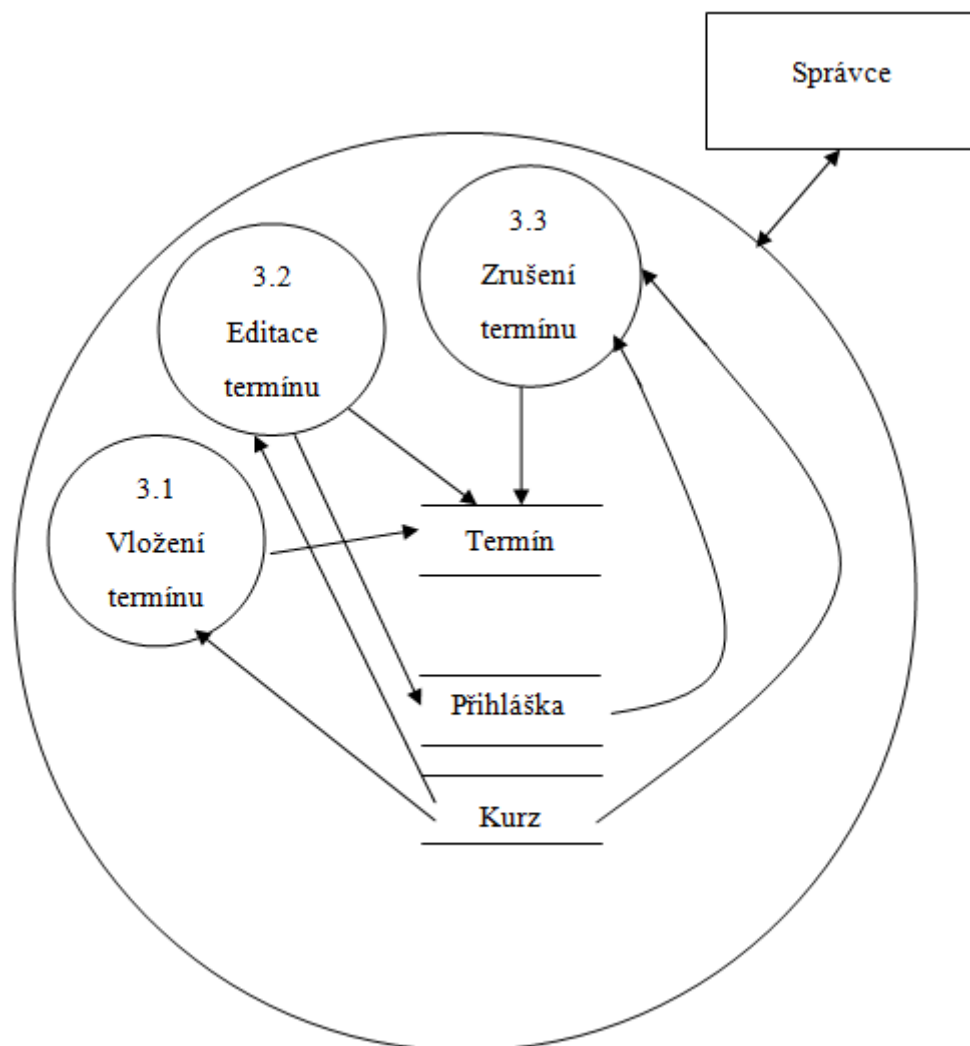
## Správa kurzů



Obrázek 6 - IS přihlašování na kurzy - DFD 1. úrovně - správa kurzů

DFD 1. úrovně – Správa kurzů se zabývá akcemi vkládání, editace a mazání kurzů. Kurz nebude moci být smazán, obsahuje-li nějaký přihlášku. Tyto operace bude provádět pouze správce IS.

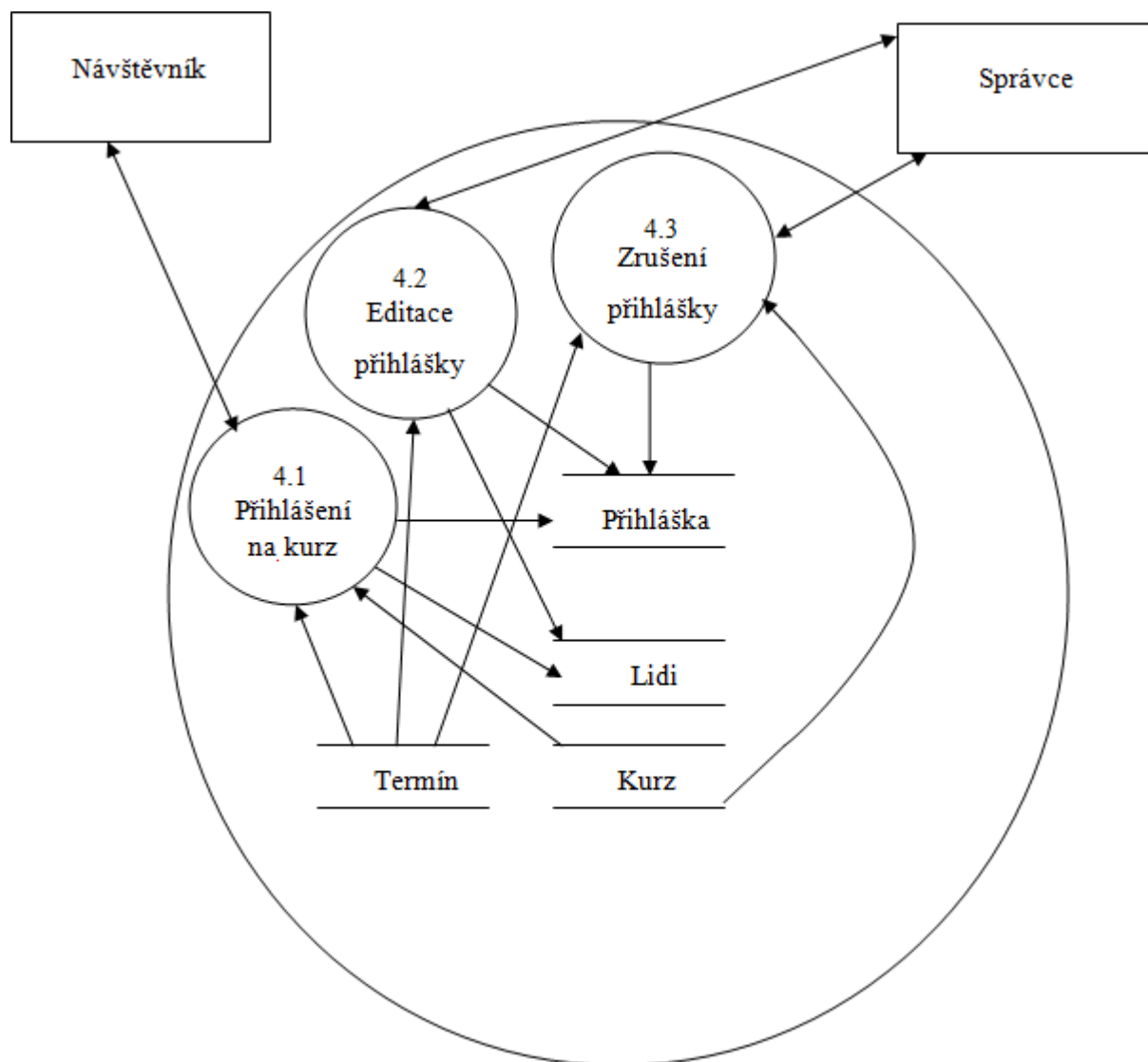
## Správa termínů



Obrázek 7 - IS přihlašování na kurzy - DFD 1. úrovně - správa termínů

DFD 1. úrovně – Správa termínů se zabývá akcemi vkládání, editování a mazání termínu. Termín může být smazán pouze v případě, že na něj není podána žádná přihláška. Termín v rámci kurzu může být dvojího typu, a to termín, klasické datum nebo jiný termín, který může být například typu „na požádání“, „kdykoliv“ apod. Jestliže bude změněn termín kurzu, bude provedena automaticky i změna termínu na přihlášce. Tyto operace smí provádět pouze správce IS.

## Správa přihlášek



Obrázek 8 - IS přihlašování na kurzy - DFD 1. úrovně - správa přihlášek

DFD 1. úrovně – Správa přihlášek se zabývá akcemi přihlášení na kurzy, což je prováděno návštěvníkem stránek, dále pak editací přihlášky (jejího stavu, viz. kapitola 2.3.4 Časová analýza) a mazáním přihlášek, což má na starost správce IS.

### 2.3.3 Minispecifikace

Minispecifikace je popis elementární funkce. Popisuje podrobně její algoritmus. Funkce na vyšších úrovních nemá smysl specifikovat, protože jsou jen množinou funkcí nižší úrovně [10, str. 32].

#### 2.3.3.1 Vložení nové kategorie (1.1)

1. Zobraz formulář pro zadání nové kategorie s tlačítkem „Vytvořit“



Kategorie:

Obrázek 9 - Formulář pro vložení kategorie

2. Načti z databáze, všechny řádky z tabulky Kategorie a zobraz „seznam kategorií“  
ID: Název:  
1 Kategorie\_1  
. . . . .
3. Správce zadá novou kategorii
4. Ulož hodnotu do proměnné Pkategorie
5. JESTLIŽE správce zmáčkne tlačítko „Vytvořit“
  - a) Načti z tabulky Kategorie řádek tabulky, kde název kategorie se rovná Pkategorie a ulož do proměnné dot
  - b) PAK VYBER PŘÍPAD  
PŘÍPAD A: JESTLIŽE název kategorie je prázdný  
PAK vypiš chybové hlášení „Nezadali jste název kategorie“ a jdi na bod 2  
PŘÍPAD B: JESTLIŽE bude proměnná dot větší jak 0  
PAK vypiš chybové hlášení „Kategorie již existuje“ a jdi na bod 2  
JINAK vygeneruj nové id\_kategorie a zapiš novou kategorii do tabulky Kategorie a vypiš hlášení „Vložena nová kategorie“ a jdi na bod 1

#### 2.3.3.2 Editování kategorie (1.2)

1. Zobraz formulář pro editování existující kategorie s tlačítkem „Uložit změny“



Kategorie:

Obrázek 10 – Formulář pro editování kategorie

2. Načti z databáze, všechny řádky z tabulky Kategorie a zobraz „seznam kategorií“ s ovládacím tlačítkem „Editovat“

ID: Název:

1 Kategorie\_1

Editovat

3. Správce stiskne tlačítko Editovat u kategorie, kterou chce editovat
4. Načti z databáze z tabulky Kategorie id kategorie do proměnné hidIdKat, název kategorie do proměnné Pkategorie a zobraz vybranou kategorii do pole kategorie
5. Administrátor provede změny
6. Ulož hodnotu do proměnné Pkategorie
7. JESTLIŽE správce zmáčkne tlačítko „Uložit změny“
8. Načti z databáze řádek tabulky s id kategorie = hidIdKat.
9. JESTLIŽE se proměnná kategorie nerovná názvu kategorie načtené z databáze s id kategorie hidIdKat
- a) JESTLIŽE název kategorie je prázdný  
PAK vypiš chybové hlášení „Musíte zadat název Kategorie!“ a jdi na bod 2
- JINAK a) zjisti počet řádku v databázi v tabulce Kategorie, kde název kategorie = Pkategorie.  
b) JESTLIŽE počet vrácených řádků bude rovno 0  
PAK vlož proměnnou kategorie do tabulky Kategorie s id kategorie hidIdKat  
JINAK vypiš chybové hlášení „Kategorie již existuje“ a jdi na bod 2

### 2.3.3.3 Smazání kategorie (1.3)

1. Načti z databáze, všechny řádky z tabulky Kategorie a zobraz „seznam kategorií“ s ovládacím tlačítkem „Smazat“

ID: Název:

1 Kategorie\_1

Smazat

2. JESTLIŽE správce zmáčkne tlačítko „Smazat“

PAK

- a) načti z databáze počet řádku z tabulky Kurz s danou id kategorie do proměnné Pcount
- b) JESTLIŽE je proměnná Pcount větší jak 0  
PAK vypiš hlášku „Kategorii nelze smazat, obsahuje kurzy!“  
JINAK smaž řádek z tabulky Kategorie s danou id kategorie



#### 2.3.3.4 Vložení nového kurzu a termínu (2.1, 3.1)

1. Zobraz formulář pro vložení nového kurzu a termínu s ovládacím tlačítkem „Vytvořit kurz“

Kategorie:	<input type="text"/>				
Název:	<input type="text"/>	Garant:	<input type="text"/>	Forma kurzu:	<input type="text"/>
Délka kurzu:	<input type="text"/>	Počet hodin:	<input type="text"/>	Ukončení kurzu:	<input type="text"/>
Pracoviště:	<input type="text"/>	Datum:	<input checked="" type="radio"/> Kdykoliv <input type="radio"/> Datum <input type="text"/>	Kapacita:	<input type="text"/>
Popis:	<input type="text"/>		Určeno pro:	<input type="text"/>	
Požadavky:	<input type="text"/>		Obsah st. balíku:	<input type="text"/>	
Průběh kurzu:	<input type="text"/>		Zisky kurzu:	<input type="text"/>	
<div>Uložit změny Vytvořit kurz</div>					

Obrázek 11 – Formulář pro vložení kurzu a termínu

2. Správce vyplní všechny položky formuláře a stiskne tlačítko „Vytvořit kurz“
3. JESTLIŽE formulář obsahuje chyby

PAK vypiš chybová hlášení a jdi na 2

JINAK

JESTLIŽE je jako Termín vybráno „Datum“

PAK

- a) vlož do proměnné Pzahajeni\_terminu data z pole Datum a do proměnné Pjiny\_termin vlož null
- b) data z formuláře a proměnné Pzahajeni\_terminu a Pjiny\_termin vlož jako nový řádek do tabulek Kurz a Termín a vypiš hlášku „Kurz vytvořen“ a vrať se na bod 1

JINAK

- a) vlož do proměnné Pzahajeni\_terminu null a do proměnné Pjiny\_termin vlož „Kdykoliv“
- b) data z formuláře a proměnné Pzahajeni\_terminu a Pjiny\_termin vlož jako nový řádek do tabulek Kurz a Termín a vypiš hlášku „Kurz vytvořen“ a vrať se na bod 1

#### 2.3.3.5 Editování kurzu / termínu (2.2, 3.2)

1. Načti z databáze všechny řádky z tabulky Kurz a Termín do seznamu kurzů a termínů s tlačítkem „Editovat“

**Kurz: Termín: Garant: Pracoviště: Kapacita: Cena:**

Kurz1 Termín1 Garant1 Pracoviště1 20 2000

Smazat

Kurz1 Termín2 Garant2 Pracoviště1 30 1500

Editovat

2. Správce vybere kurz (termín), který chce editovat a zmáčkne tlačítko „Editovat“
3. Načti z databáze příslušný řádek z tabulky Kurz a Termín podle id kurzu a vlož hodnoty do formuláře „vytvoření kurzu“ s tlačítkem „Uložit změny“

Kategorie:	<input type="text"/>		Garant:	<input type="text"/>	Forma kurzu:	<input type="text"/>
Název:	<input type="text"/>					
Délka kurzu:	<input type="text"/>	Počet hodin:	<input type="text"/>	Ukončení kurzu:	<input type="text"/>	Cena:
Pracoviště:	<input type="text"/>		Datum:	<input checked="" type="radio"/> Kdykoliv <input type="radio"/> Datum	<input type="text"/>	Kapacita:
Popis:	<input type="text"/>		Určeno pro:	<input type="text"/>		
Požadavky:	<input type="text"/>		Obsah st. balíku:	<input type="text"/>		
Průběh kurzu:	<input type="text"/>		Zisky kurzu:	<input type="text"/>		

**Obrázek 12 – Formulář pro editování kurzu či termínu**

4. Správce změní hodnoty polí, které potřebuje a stiskne tlačítko „Uložit změny“
5. JESTLIŽE formulář obsahuje chyby  
PAK vypiš chybová hlášení a jdi na 2  
JINAK změn hodnoty změněné správcem ve formuláři pro příslušný řádek tabulky Kurz a Termín

#### 2.3.3.6 Smazání kurzu / termínu (2.3, 3.3)

Načti z databáze všechny řádky z tabulky Kurz a Termín do seznamu kurzů a termínů s tlačítkem „Smazat“

**Kurz: Termín: Garant: Pracoviště: Kapacita: Cena:**

Kurz1 Termín1 Garant1 Pracoviště1 20 2000

Smazat

Kurz1 Termín2 Garant2 Pracoviště1 30 1500

Smazat

.....

1. Správce vybere kurz (termín), který chce smazat a zmáčkne tlačítko „Smazat“
2. Načti z databáze počet řádků z tabulky Přihláška s příslušným id kurzu
3. JESTLIŽE je počet řádků v tabulce Přihláška s daným id kurzu větší jak 0  
PAK vypiš hlášku „Kurz nelze smazat, jsou na něj vázány přihlášky!“  
JINAK smaž z tabulky Kurz a Termín řádek s daným id kurzu

### 2.3.3.7 Vložení termínu k existujícímu kurzu (3.2)

1. Načti z databáze všechny řádky z tabulky Kurz a Termín do seznamu kurzu a termínů s tlačítkem „Nový termín“

**Kurz: Termín: Garant: Pracoviště: Kapacita: Cena:**

Kurz1 Termín1 Garant1 Pracoviště1 20 2000

Nový termín

Kurz1 Termín2 Garant2 Pracoviště1 30 1500

Nový termín

.....

2. Správce vybere kurz, ke kterému chce vložit nový termín a zmáčkne tlačítko „Nový termín“
3. Načti z databáze příslušný řádek z tabulky Kurz a Termín a vlož hodnoty do formuláře „pro vytvoření kurzu“ s tlačítkem „Vložit termín“, kromě kapacita, pracoviště, datum, termín a nastav je na disabled
4. Správce vloží nový termín a zmáčkne tlačítko „Vložit termín“
5. JESTLIŽE formulář obsahuje chyby  
PAK vypiš chybová hlášení a jdi na 2  
JINAK vlož do tabulky Termín podle id kurzu nový řádek a vypiš hlášku „Nový termín vložen“

### 2.3.3.8 Vyplnění a odeslání přihlášky ke studiu (4.1)

1. Načti informace z databáze o kategoriích, kurzech a termínech.
2. Zobraz uživateli přihlášku ke studiu s položkami a tlačítkem „Odeslat přihlášku“
3. a) Zapláceno bude nastaveno na false  
b) Ulož aktuální datum do proměnné Pdatum\_podání.
4. JESTLIŽE formulář obsahuje chyby  
PAK vypiš chybová hlášení a jdi na 2
5. JESTLIŽE bude počet již podaných přihlášek na tento kurz v tomto termínu roven kapacitě tohoto kurzu v tomto termínu  
PAK upozorni uživatele, že je již kapacita kurzu naplněna a vrať se na bod 1  
JINAK vlož nový záznam do tabulky Přihláška a Lidi a vypiš hlášku „Přihláška přijata!“

Kurz:	<input type="text"/>
Termín:	<input type="text"/>
Jméno:(*)	<input type="text"/>
Příjmení:(*)	<input type="text"/>
Heslo:(*)	<input type="text"/>
Heslo znovu:(*)	<input type="text"/>
Ulice:(*)	<input type="text"/>
Město:(*)	<input type="text"/>
PSČ:(*)	<input type="text"/>
Telefon:(*)	<input type="text"/>
Email:(*)	<input type="text"/>
Zaměstnavatel:	<input type="text"/>
Způsob úhrady:(**)	Osobně <input checked="" type="radio"/> Na účet <input type="radio"/> Poštou: <input type="radio"/>
Poznámka:	<input type="text"/>
	<input type="button" value="Odeslat přihlášku"/>

Obrázek 13 - Formulář pro odeslání přihlášky

#### 2.3.3.9 Smazání přihlášky (4.3)

1. Načti z databáze všechny řádky z tabulky Přihláška, Lidi, Termín do seznamu přihlášek s tlačítkem „Smazat“

Přihláška	Příjmení	Jméno	Kurz	Termín	Stav	Zaplaceno
1	Novák	Jan	Kurz1	2010-08-08	Přijatá <input type="radio"/> Zpracovaná <input type="radio"/> Ne <input type="radio"/> Ano <input type="radio"/>	
					<input type="button" value="Smazat"/>	

Obrázek 14 - Seznam s přihláškami

2. JESTLIŽE správce zmáčkne tlačítko „Smazat“  
PAK smaž přihlášku z tabulky Přihláška s daným id přihlášky

### 2.3.3.10 Zobrazení detailu přihlášky

1. Načti seznam přihlášek daného přihlášeného uživatele a zobraz je s tlačítkem „Detail“


Přihláška	Kurz	Termín	Stav	Zaplaceno	
1	Kurz1	2010-08-08			<a href="#">Detail</a>

Obrázek 15 - Seznam přihlášek uživatele

2. JESTLIŽE uživatel (správce) klikne na tlačítko „Detail“.

PAK

- a) Načti z databáze podle id přihlášky řádek z tabulky Přihláška, Kurz, Lidi, Termín
- b) Ulož hodnoty z databáze do proměnných a zobraz v přihlášce, v HTML formátu

Osobní
<b>Jméno:</b> Petr <b>Příjmení:</b> Novák
<b>Ulice:</b> Novosadská 34 <b>Město:</b> Přerov <b>PSČ:</b> 75002
<b>Telefon:</b> +420777889090 <b>email:</b> petr.novak@seznam.cz
<b>Zaměstnavatel:</b> 
Kurz
<b>Název:</b> Stavby a zařízení CO <b>Termín:</b> Kdykoliv <b>Cena:</b> 4 000,-
<b>Délka kurzu:</b> 1 semestr <b>Počet hodin:</b> 35
<b>Pracoviště:</b> oddělení 895 – Regionální centrum celoživotního vzdělávání, VŠB-TU Ostrava
Způsob úhrady
<b>Příkazem k úhradě (bezhotovostně z účtu):</b> <i>název účtu:</i> Vysoká škola báňská – Technická univerzita Ostrava <i>číslo účtu:</i> 127089559/0300 <i>variabilní symbol:</i> 895 702 <i>konstantní symbol:</i> rodné číslo ČSOB, a.s., pobočka Ostrava-Poruba Hlavní třída 703, 708 39
<b>Datum podání:</b> 2010-03-03

Obrázek 16 - Detail přihlášky

### 2.3.3.11 Přihlášení do systému

1. Zobraz uživateli položky nutné k vyplnění, které jsou potřebné pro přístup do systému
2. Uživatel vyplní položky email a heslo
3. Ulož hodnoty do proměnných Pemail a Ppassword
4. Proveď zahashování Password pomocí funkce SHA1
5. JESTLIŽE hodnota v proměnné email odpovídá záznamu v databázi, v tabulce Lidi

PAK načti z databáze, z tabulky Lidi hodnoty z tohoto řádku

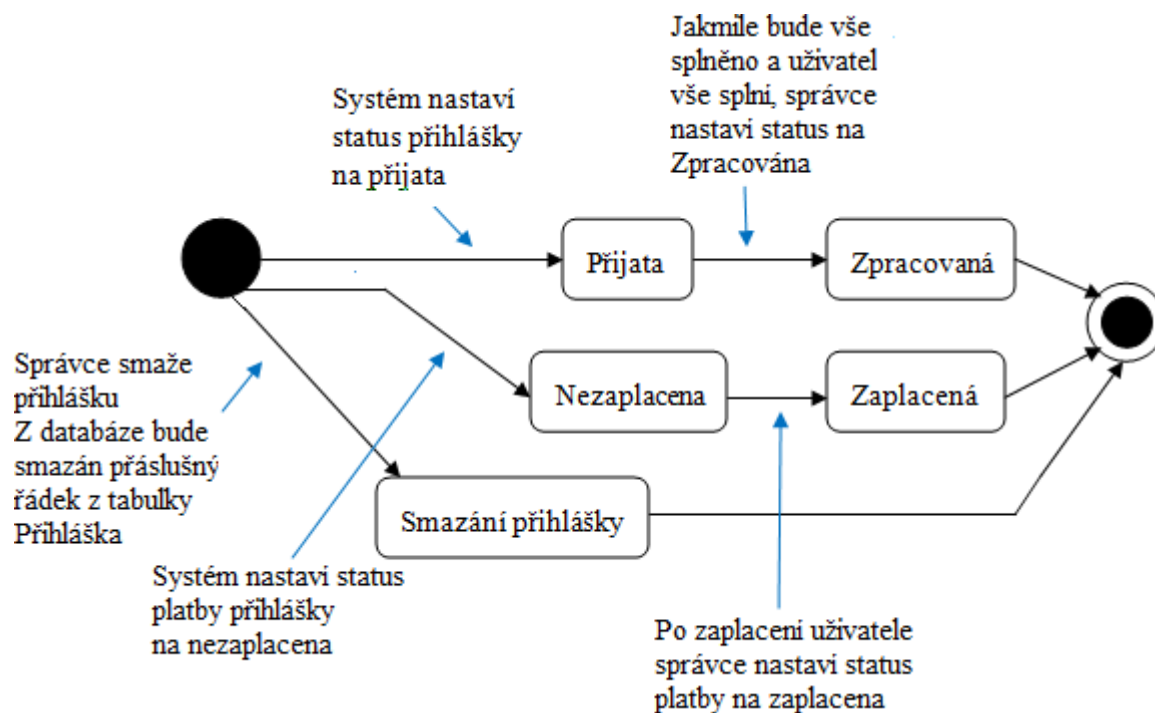
JESTLIŽE Ppassword = password

PAK pusť uživatele do systému

JINAK vypiš hlášku „Zadali jste špatný email nebo heslo“

### 2.3.4 Časová analýza stavu přihlášek

Časová analýza slouží k modelování chování systému v závislosti na čase nebo na pořadí funkcí. Popisuje časové následnosti procesů, které DFD nezaznamenává. Modeluje chování systému v závislosti na působení **vnějších událostí** nebo na základě **vnitřních změn stavů** [10, str. 40].



Obrázek 17 - Časová analýza stavu přihlášky

## 2.4 Návrh implementace

Kapitola zabývající se popisem technologií použitých pro tvorbu výsledného IS, dále pak použitým software a hardware. Dále pak obsahuje podkapitoly „Změna v datových slovnících“ popisující změny oproti původnímu návrhu a „Indexová analýza“ [10].

### 2.4.1 Použité technologie

#### 2.4.1.1 XHTML

Jedná se o značkovací jazyk pro tvorbu hypertextových dokumentů v prostředí WWW vyvinutý W3C. XHTML je nástupcem HTML založený na XML. Rozlišujeme 3 druhy:

- XHTML 1.0 Strict - čistě strukturální značkování, neobsahuje žádné značky spojené s formátováním vzhledu
- XHTML 1.0 Transitional - povoluje atributy pro formátování textu a odkazů v elementu body a některé další atributy
- XHTML 1.0 Frameset - používá se při použití rámců pro rozdělení okna prohlížeče na dvě nebo více částí

#### XHTML se od HTML liší v těchto bodech

- Všechny tagy a jejich atributy musí být zapsány malými písmeny a to z toho důvodu, že jsou takto deklarované v odkazované DTD a X(HT)ML je case sensitive.
- Všechny tagy musí být ukončené a to včetně nepárových jako jsou <meta>, <img> apod. př. <img ..... />
- Všechny hodnoty atributů musí být v uvozovkách
- Dokument musí začínat XML deklarací. Její použití není povinné, pokud je dokument kódován v UTF-8 nebo pokud určujeme kódování vyšší protokolem (http například)
- Pokud potřebujeme pracovat s rámy, můžeme deklarovat XHTML 1.0 Frameset a pro jednotlivé stránky XHTML 1.0 Transitional.

Více podrobností zde [8].

#### 2.4.1.2 CSS

V dnešní době nepostradatelný pomocník při tvorbě webových stránek. Kaskádové styly definují vzhled www stránek odděleně od jeho struktury a obsahu. Opět se jedná o jazyk z dílny W3C. Byly vydány dvě úrovně specifikace CSS1 a CSS2, dokončuje se revize CSS 2.1 a pracuje se na verzi CSS3.

#### Umístění stylů:

- V samostatném souboru , který importujeme v hlavičce:

```
<link rel="stylesheet" type="text/css" href="styles.css" media="screen" />
```

- Přímě v hlavičce HTML dokumentu:

```
<style type="text/css">
  body {
    font-family: Verdana;
    font-size: 13px;
    padding:0px;
    margin:0px;    }
</style>
```

- Přímou v elementu (pomocí atributu style):

```
<td style="background-image:url(../images/header-3.jpg)">
<b>Jméno:</b></td>
```

#### **Vazby mezi stylem a elementy:**

- Podle jména elementu
- Podle identifikátoru elementu (id)
- Podle třídy elementu (class)
- Podle strukturálního umístění v dokumentu (zanořování a dědičnost)

#### **2.4.1.3 Javascript**

Jedná se o skriptovací programovací jazyk, který řeší dynamiku webových stránek na straně klienta. Je podobný jazykům C/C++, Java. JS je přímou součástí zdrojového kódu a je prováděn až ve chvíli potřeby, je závislý na interpretačním prostředí (prohlížeč), objektově orientovaný, ale beztřídní (prototypy), je case-sensitive atd.

JS umožňuje ovlivňování vzhled a obsah dokumentů, manipulovat s obrázky a jinými prvky, částečně řídit a ovlivňovat prohlížeč, provádět algoritmy, matematické výpočty zpracovávat události od uživatele, ukládat a číst Cookies atd. .[7]

#### **Umístění Javascriptu:**

- V samotném souboru , který importujeme v hlavičce:

```
<script language="javascript" src="js.js">?</script>
```

- Přímý zápis kódu v dokumentu

```
<script language="javascript"> .....kód.....</script>
```



#### 2.4.1.4 PHP

**Skriptovací programovací jazyk**, určený především pro programování dynamických internetových stránek. Nejčastěji se začleňuje přímo do struktury jazyka HTML, XHTML, což lze využít při tvorbě webových aplikací.

PHP skripty jsou většinou prováděny na straně serveru, k uživateli je přenášen až výsledek jejich činnosti. Syntaxe jazyka je inspirována několika programovacími jazyky (Perl, C, Pascal a Java). PHP je nezávislý na platformě, skripty fungují bez větších úprav na různých operačních systémech. Podporuje mnoho knihoven pro různé účely - např. zpracování textu, grafiky, práci se soubory, přístup k většině databázových systémů (MySQL, Oracle aj.), podporuje celou řadu internetových protokolů (HTTP, POP3, FTP aj.).

Většinou se používá v kombinaci s operačním systémem Linux, databázovým systémem MYSQL a webovým serverem Apache => využíván k tvorbě webových aplikací. Pro tuto kombinaci se vžila zkratka LAMP

#### Výhody PHP

- Rozsáhlý soubor funkcí v základní knihovně PHP, další funkce v PECL.
- Nativní podpora mnoha databázových systémů.
- Multiplatformní
- Možnost využití nativních funkcí operačního systému (možná nekompatibilita s jiným OS).<sup>[4]</sup>
- Obrovská podpora na hostingových službách.
- Rozsáhlá dokumentace.

#### Nevýhody PHP

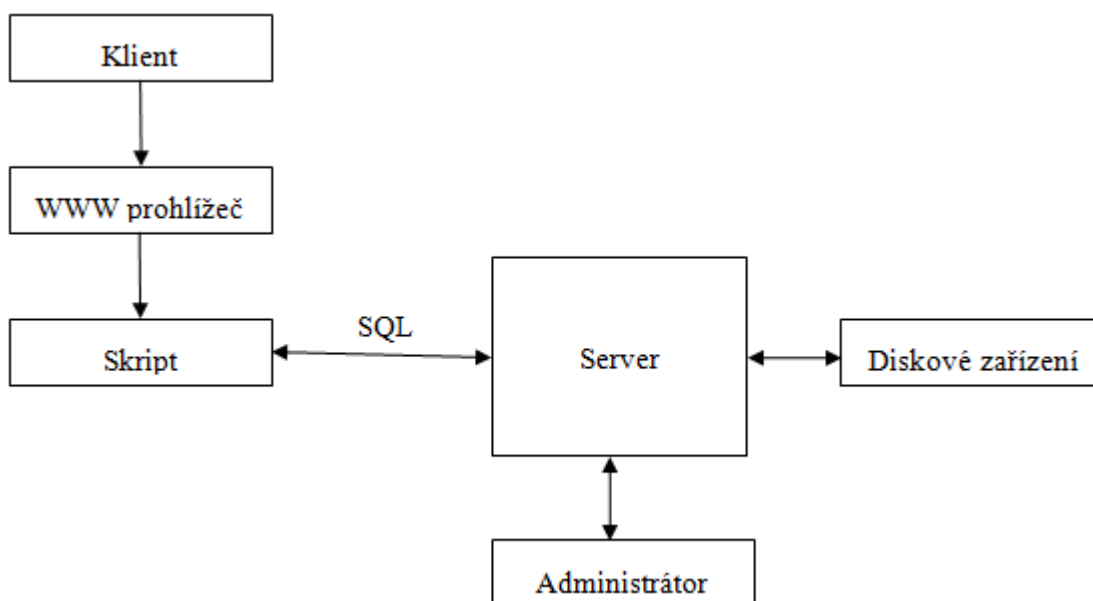
- Jazyk PHP není nikde definován, je popsán pouze jeho implementací.
- Mírně nekonzistentní vývoj v minulosti, který si s sebou PHP nese dosud (někdy často měnící se příkazy či jejich parametry atd...).
- Nekonzistentní pojmenování funkcí, např.:
  - strpos(), strchr(), ale str\_replace(), str\_pad().
- Ačkoliv jazyk podporuje výjimky, jeho knihovna je používá jen zřídka.
- Slabší podpora Unicode, pouze přes PHP knihovnu (v PHP 6 má být Unicode řetězec jako základní typ).
- Ve standardní distribuci chybí ladící nástroj.
- Po zpracování požadavku neudrží kontext aplikace, vytváří jej vždy znovu (oslabuje výkon).

Více podrobností zde [5].

#### 2.4.1.5 MySQL

Jedná se o databázový systém, vytvořený švédskou firmou MySQL AB, nyní vlastněný společností Sun Microsystems, dceřinnou společností Oracle Corporation. Je považován za průkopníka dvojího licencování – je k dispozici jak pod bezplatnou licenci GPL, tak pod komerční placenou licenci. MySQL je multiplatformní relační databáze. Komunikace s ní probíhá pomocí jazyka SQL. Podobně jako u ostatních SQL databází se jedná o dialekt tohoto jazyka s některými rozšířeními. Pro svou snadnou implementovatelnost (lze jej instalovat na různé operační systémy), výkon a díky tomu, že se jedná o volně šiřitelný software, má vysoký podíl na v současnosti používaných databázích. MySQL bylo od počátku optimalizováno především na rychlost, a to i za cenu některých zjednodušení: má jen jednoduché způsoby zálohování, a až donedávna nepodporovalo pohledy, trigger, a uložené procedury. Tyto vlastnosti jsou doplňovány teprve v posledních letech. [6]

#### 2.4.2 Jak komunikují SŘBD



Obrázek 18 – Jak komunikují SŘBD

Klient prostřednictvím prohlížeče odešle skript s sql dotazem na server odkud obratem dostane výsledek této činnosti a je mu zobrazena výsledná www stránka.

#### 2.4.3 Hardware, Software

Oba informační systémy byly vyvíjeny v textovém editoru PSPad, který je bezplatně licencován. Jedná se o jednoduchý editor, jehož hlavní výhodou je jeho jednoduchost, rychlost a přehlednost. Dále PSPad umožňuje nastavení kompilátorů pro programovací jazyky, s možností jejich výstupu do logovacího okna integrovaného v prostředí editoru. Program podporuje přes 30 různých prostředí.

### Další použitý software:

**VertigoServ:** tento software byl vyvinut, jako dostupný, vysoce profesionální a jednoduše instalovatelný balík, skládající se z Apache, PHP, MySQL, PhpMyAdmin (nástroj napsaný v jazyce PHP pro správu administrace MySQL) a dalších nástrojů pro platformu Windows.

### Aktuální verze 2.21 obsahuje tyto nástroje:

- Apache 2.0.63
- PHP 5.2.6,
- MySQL 5.0.51b, SQLite 3.5.9
- PhpMyAdmin 2.11.7

Informační systémy byly odzkoušeny na notebooku ASUS: Procesor: 2,0 GHz (AMD Turion 64 X2) Paměť: 1024 MB (PC5300 SO-DIMM 200pin (DDR2-667)) Hard disk: 160 GB, 5400 RPM. Operační systém: Windows Vista.

### 2.4.4 Změna v datových slovnících

Tabulka: **Přihláška**

Datový typ u atributu **stav** a **zaplaceno** byl změněn z datového typu **boolean**, který nabývá hodnot true nebo false na datový typ **TINYINT**, což je celočíselný datový typ, který může nabývat hodnot od -128 do 127. My využijeme hodnotu 0, která nám nahradí původní hodnotu false a hodnotu 1, která nám nahradí původní hodnotu true. Datový typ byl nahrazen z důvodu toho, že datový typ **Boolean** se v MySQL databázi nevyskytuje.

### 2.4.5 Indexová analýza

U **indexové analýzy** budeme na základě jednotlivých minispecifikací zvažovat, u kterých hodnot v daných tabulkách vytvoříme indexy. Projdeme tedy minispecifikace, které pracují s databázovými tabulkami a zvažíme zda se atributy tabulek využijí pro některé z následujících činností:

- hledání
- používání setříděný seznamů
- spojení tabulek
- ověřování jednoznačností v téže tabulce
- ověřování existence v jiné tabulce

Indexy byly vytvořeny pro všechny primární klíče, pro atributy, které spojují tabulky, tj. cizí klíče a u atributu podle kterých jsou tříděny různé seznamy nebo využito vyhledávání. U všech těchto atributů byly použity udržované indexy z důvodu jejich neustálého používání.[10, str. 63]

### Kategorie

název	typ	vel.	klíč	null	index	důvod indexu
id_kategorie	int	5	PK	N	udržovaný	jednoznačná identifikace kategorie
název_kategorie	varchar	40	N	N	udržovaný	vyhledávání, seříděný výstup

### Kurz

id_kurzu	int	10	PK	N	udržovaný	jednoznačná identifikace kurzu
název_kurzu	varchar	50	N	N	udržovaný	vyhledávání, seříděný výstup
id_kategorie	int	5	FK	N	udržovaný	jednoznačná identifikace kategorie, cizí klíč z tabulky Kategorie

### Termín

id_termínu	int	5	PK	N	udržovaný	jednoznačná identifikace termínu
zahájení_termínu	date		N	N	udržovaný	vyhledávání, seříděný výstup
jiný_termín	var- char	30	N	N	udržovaný	vyhledávání, seříděný výstup
id_kurz	int	5	FK	N	udržovaný	jednoznačná identifikace kurzu, cizí klíč z tabulky Kurz

### Přihláška

id_přihlášky	int	5	PK	N	udržovaný	jednoznačná identifikace přihlášky
id_kurzu	int	5	FK	N	udržovaný	jednoznačná identifikace kurzu, cizí klíč z tabulky Kurz
id_termínu	int	5	FK	N	udržovaný	jednoznačná identifikace termínu, cizí klíč z tabulky Termín
id_lidi	int	5	FK	N	udržovaný	jednoznační identifikace lidí, cizí klíč z tabulky Lidi

## Lidi

id_lidi	int	5	PK	N	udržovaný	jednoznačná identifikace Lidi
příjmení	varchar	20	N	N	udržovaný	vyhledávání, setříděný výstup
id_role	int	5	FK	N	udržovaný	jednoznačná identifikace role, cizí klíč z tabulky Role

## Role

id_role	int	5	PK	N	udržovaný	jednoznačná identifikace role
název_role	varchar	15	N	N	udržovaný	vyhledávání, setříděný výstup

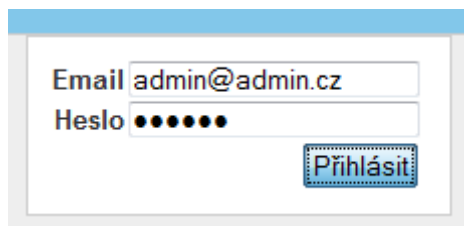
## 2.5 Popis implementace

Tato kapitola se skládá z podkapitol uživatelské příručky, programátorská příručky a popisu implementace.

### 2.5.1 Uživatelská příručka

Kapitola uživatelská příručka se zabývá popisem IS, jeho funkcí a možností činnosti s IS ze strany uživatelů používající IS a to jak správce systému, tak běžného uživatele.

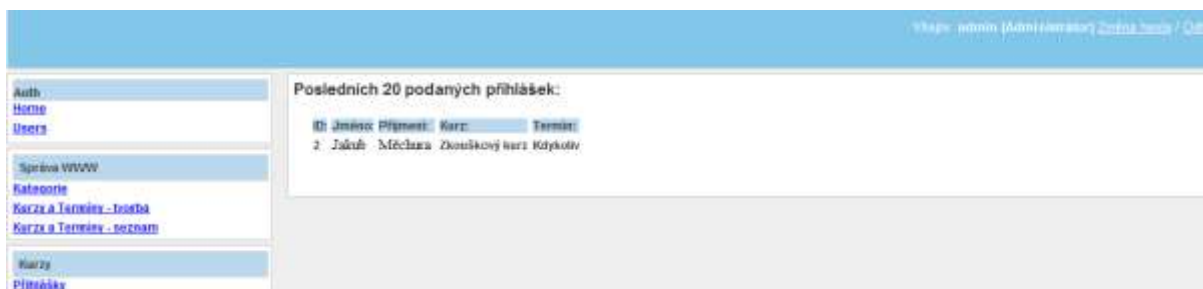
#### 2.5.1.1 Přihlášení



Obrázek 19 - IS přihlašování na kurzy - přihlášení do systému

1. Uživatel zadá svůj email, heslo a stiskne tlačítko „přihlásit“.

### 2.5.1.2 Administrační rozhraní – úvodní obrazovka



Obrázek 20 - IS přihlašování na kurzy - úvodní obrazovka

### 2.5.1.3 Administrační rozhraní – Users

Obrázek 21 - IS přihlašování na kurzy - sekce Users

1. Správce má možnost editovat, mazat a popř. i vkládat uživatele.
2. Pro urychlení práce může využít vyhledávání podle příjmení.
3. V případě, že správce chce vložit nového uživatele (např. nového správce) vyplní všechny položky a zmáčkne tlačítko „Vytvořit uživatele“.
4. V případě, že bude chtít editovat údaje o uživateli, vybere příslušného uživatele a zmáčkne tlačítko se symbolem tužky. Změní údaje a zmáčkne tlačítko „Uložit změny“.
5. Jestliže správce bude chtít smazat uživatele má dvě možnosti:
  - a) Zmáčknutím tlačítka se symbolem křížku.

- b) Využití „skupinového mazání“, kdy zatrhne všechny uživatele, které bude chtít smazat, vybere nad seznamem uživatelů možnost „Smazat uživatele“ a zmáčkne tlačítko „Provést“.

V obou případech nelze smazat uživatele, na které je vázána přihláška.

#### 2.5.1.4 Administrační rozhraní – Kategorie

Obrázek 22 - IS přihlašování na kurzy - sekce kategorie

1. Vkládání, editování a mazání probíhá stejným způsobem jako v kategorii „Users“.
2. Smazána nemůže být kategorie, ke které je vázán kurz.

#### 2.5.1.5 Administrační rozhraní – Kurzy a Termíny – tvorba, seznam

Obrázek 23 – IS přihlašování na kurzy - kurzy a termíny tvorba

Kurz:	Garant:	Termín:	Pracoviště:	Kapacita:	Cena:
Stavby a zařízení CO ng. Jaroslav Kovařík	Kdykoliv		oddělení 895 – Regionální centrum celoživotního vzdělávání, VŠB-TU Ostrava	30	4 000,-

Obrázek 24 - IS přihlašování na kurzy - kurzy a termíny seznam

1. Vkládání, editování a mazání probíhá jako v předchozích dvou případech.
2. U vkládání a editování je třeba rozlišovat jaký zvolíme Termín. Jestliže zvolíme termín kdykoliv, do nadcházejícího textového pole už nic nevyplňujeme. Ovšem jestliže zvolíme možnost Datum, do nadcházejícího textového pole zadáme datum ve formátu YYYY-MM-DD.
3. Nelze smazat kurz, ke kterému jsou vázány přihlášky.
4. Pokud budeme chtít vložit termín k již existujícímu kurzu, stiskneme u v seznamu kurzu u příslušného kurzu symbol „kalendáře“. Poté do vyznačených polí vepíšeme údaje o novém termínu.
5. Lze možno vyhledávat kurz podle je názvu => Combo box v seznamu kurzů.




#### 2.5.1.6 Administrační rozhraní – přihláška

Kurz	Termín	Přihlášených
Zkouškový kurz	Kdykoliv	<a href="#">2</a>

Obrázek 25 - IS přihlašování na kurzy - sekce přihláška

1. V kategorii přihlášky je možno vyhledávat podle kurzu a to výběrem dané položky z combo boxu.
2. V seznamu jsou sloupce Kurz, Termín a počet přihlášek v kurzu.
3. Počet přihlášených slouží jako odkaz na seznam lidí přihlášených v kurzu.
4. Po kliknutí správce na toto číslo (odkaz) se mu zobrazí následující seznam



Seznam přihlášek kurzu: Zkouškový kurz									
Seznam									
Přihláška	Příjmení	Jméno	Kurz	Termín	Stav	Zaplaceno			
1	Měchura	Jakub	Zkouškový kurz	Kdykoliv	Přijata	<input checked="" type="radio"/> Zpracovná	<input type="radio"/> Nezaplacena	<input checked="" type="radio"/> Zaplacená	  

Obrázek 26 - IS přihlašování na kurzy - sekce přihláška - seznam přihlášek v kurzu

- V tomto seznamu vedle každého řádku máme tři ikony
  - Ikona **diskety** – slouží k uložení změny sloupce Stav a Zaplaceno
  - Ikona **křížek** – jako u předchozích sekcí slouží ke smazání přihlášky.
  - Ikona **list** – slouží k zobrazení detailu přihlášky, viz. **obrázek 31 – detail přihlášky**.
- Nad tímto seznamem se nachází tlačítko „Seznam“, které slouží k zobrazení seznamu lidí v kurzu pro tisk.

Účastníci kurzu: Stavby a zařízení CO			Termín: Kdykoliv
Přihláška č.:	Příjmení:	Jméno:	Zaplaceno:
37	Novák	Petr	NE

Obrázek 27 - IS přihlašování na kurzy – seznam lidí v kurzu pro tisk

#### 2.5.1.7 Administrační a uživatelské rozhraní – změna hesla

Změna hesla	
Staré heslo:	<input type="password"/>
Nové heslo:	<input type="password"/>
Nové heslo znovu:	<input type="password"/>
	<input type="button" value="Změnit"/>

Obrázek 28 - IS přihlašování na kurzy - změna hesla

- Uživatel zadá staré heslo, heslo nové a jeho potvrzení a zmáčkne tlačítko „Změnit“

**Pozn.** U všech seznamů, kde jsou sloupce označeny jako odkaz, lze tyto seznamy seřadit podle daného sloupce po kliknutí na daný odkaz sloupce.

### 2.5.1.8 Administrační a uživatelské rozhraní – odhlášení

Vítejte: petr.novak@seznam.cz (Uživatel) [Změna hesla](#) / [Odhlásit](#)

Obrázek 29 - IS přihlašování na kurzy - odhlášení

1. Po stisknutí „Odhlásit“ bude uživatel odhlášen.

### 2.5.1.9 Uživatelské rozhraní



Obrázek 30 - IS přihlašování na kurzy - uživatelské rozhraní

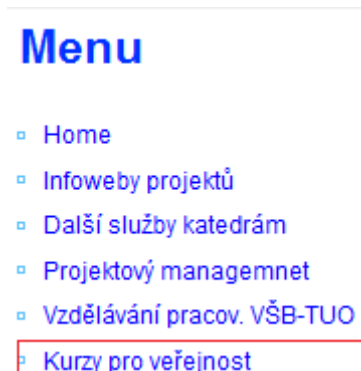
1. Po přihlášení se uživateli zobrazí seznam jeho přihlášek. U každé přihlášky je vidět, zda je zaplácena, jaký je její stav a tlačítko „Prohlédnout“. Po stisknutí tlačítka se zobrazí přihláška, viz **obrázek 31 – detail přihlášky**

Osobní	
<b>Jméno:</b>	Petr <b>Příjmení:</b> Novák
<b>Ulice:</b>	Novosadská 34 <b>Město:</b> Přerov <b>PSČ:</b> 75002
<b>Telefon:</b>	+420777889090 <b>email:</b> petr.novak@seznam.cz
<b>Zaměstnavatel:</b>	
Kurz	
<b>Název:</b>	Stavby a zařízení CO <b>Termín:</b> Kdykoliv <b>Cena:</b> 4 000,-
<b>Délka kurzu:</b>	1 semestr <b>Počet hodin:</b> 35
<b>Pracoviště:</b>	oddělení 895 – Regionální centrum celoživotního vzdělávání, VŠB-TU Ostrava
Způsob úhrady	
<b>Příkazem k úhradě (bezhotovostně z účtu):</b>	
<i>název účtu:</i>	Vysoká škola báňská – Technická univerzita Ostrava
<i>číslo účtu:</i>	127089559/0300
<i>variabilní symbol:</i>	895 702
<i>konstantní symbol:</i>	rodné číslo
	ČSOB, a.s., pobočka Ostrava-Poruba Hlavní třída 703, 708 39
<b>Datum podání:</b>	2010-03-03

Obrázek 31 - IS přihlašování na kurzy - detail přihlášky

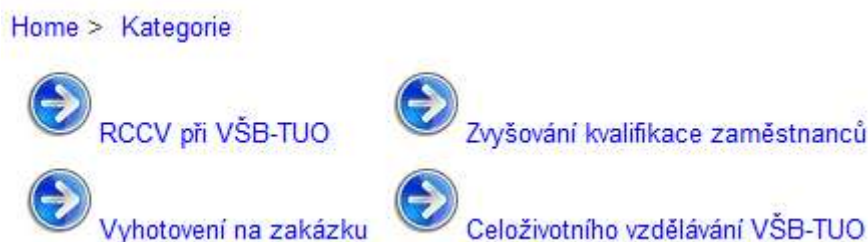
#### 2.5.1.10 Podávání přihlášky

1. Na stránce [homel.vsb.cz/~mec063/bak](http://homel.vsb.cz/~mec063/bak) klikne uživatel na sekci „Kurzy pro veřejnost“.



Obrázek 32 - WWW stránky – menu

2. V rámci této sekce jsou určité kategorie.



Obrázek 33 WWW stránky – kurzy pro veřejnost

3. V rámci kategorii je seznam kurzu.

Home > Kategorie > RCCV při VŠB-TUO

ID:	Název kurzu:	
2	Zkouškový kurz	<a href="#">Detail</a>
3	Krizové řízení	<a href="#">Detail</a>
4	Všeobecná bezpečnost	<a href="#">Detail</a>

Obrázek 34 - WWW stránky - seznam kurzů v rámci kategorie

4. Po kliknutí na odkaz „Detail“ se zobrazí detail o kurzu. V rámci tohoto detailu je seznam termínů, včetně možnosti podání přihlášky.

Home > Kategorie > RCCV při VŠB-TUO > Všeobecná bezpečnost

### Všeobecná bezpečnost

**Garant:** doc. Ing. Ivana Bartlová, CSc.

**Pracoviště:**

**Popis:**

Kurz je přípravou k získání Odborné způsobilosti k bezpečnosti a ochraně zdraví při práci. Cílem tohoto modulu je poskytnout studentům maximální množství informací z oblasti požární ochrany, bezpečnosti práce, zacházení s nebezpečnými a hořlavými látkami, managementem rizika až po krizové plánování.

.....

**Cena modulu:** 4 000

Termín	Kapacita	Cena	Místo	
Kdykoliv 10	4 000		oddělení 895 – Regionální centrum celoživotního vzdělávání VŠB-TU Ostrava	<a href="#">Přihláška</a>

Obrázek 35 - WWW stránky - detail o Kurzu

5. Po kliknutí na odkaz „Přihláška“ se zobrazí přihláška.

[Kategorie](#) > [RCCV při VŠB-TUO](#) > [Všeobecná bezpečnost](#) > [Přihláška](#)

### Přihláška pro přihlášení do kurzu

Kurz:	<input type="text" value="Všeobecná bezpečnost"/>
Termín:	<input type="text" value="Kdykoliv"/>
Jméno:(*)	<input type="text"/>
Příjmení:(*)	<input type="text"/>
Heslo:(*)	<input type="password"/>
Heslo znovu:(*)	<input type="password"/>
Ulice:(*)	<input type="text"/>
Město:(*)	<input type="text"/>
PSČ:(*)	<input type="text"/>
Telefon:(*)	<input type="text"/>
Email:(*)	<input type="text"/>
Zaměstnavatel:	<input type="text"/>
Způsob úhrady:(**)	<input checked="" type="radio"/> Osobně <input type="radio"/> Na účet <input type="radio"/> Poštou: <input type="radio"/>
Poznámka:	<input type="text"/>
	<input type="button" value="Odeslat přihlášku"/>

Obrázek 36 - WWW stránky - přihláška na kurz

6. Po odeslání přihlášky se pod přihláškou zobrazí hláška „**Přihláška odeslána!**“

## 2.5.2 Programátorská příručka

Programátorskou příručku lze najít na přiloženém cd ve složce:

**/Projekt1/Programátorská příručka**

V této složce lze najít popis jednotlivých skriptů a funkcí v HTML formátu.

## 2.5.3 Popis instalace

Na přiloženém CD ve složce:

**/Software/VertrigoServ**

Lze najít software VertrigoServ, po jehož instalaci a spuštění lze testovat výslednou aplikaci.

Po dokončení instalace a spuštění programu VertrigoServ spustíme program PhpMyAdmin ( **uživatel: root, heslo: vertrigo**), ve kterém vytvoříme databázi „**bak**“ a spustíme skript **skript.sql**, který nám vytvoří všechny potřebné tabulky včetně dat. V rámci tohoto skriptu bude vytvořen uživatel „admin“ s heslem „bbbbbb“.

Tento skript lze najít na přiloženém CD ve složce **/Projekt1/sql/**

Poté stačí do nainstalované složky **VertrigoServ/www/** zkopírovat složku **bak**, která se taktéž nachází ve složce **/Projekt1** a spustit stránku **localhost/bak**.

Výsledný IS se nachází na www stránkách <http://homel.vsb.cz/~mec063/bak/> , sekce Vaše přihlášky.



## 3 Informační systém pro evidenci projektů

### 3.1 Zadání

V této kapitole je specifikováno proč a k čemu je daný systém vyvíjen, kdo s ním bude pracovat a jaké jsou požadavky na něj, a to jak po stránce funkční, tak po stránce provozní.

#### 3.1.1 Proč?

Cílem je vytvořit informační systém (dále jen IS), do kterého budou mít přístup zaměstnanci školy pracující na jednotlivých projektech dotovaných evropskými sociálními fondy.

#### 3.1.2 K čemu?

IS vzniká za účelem snadnější komunikace mezi členy pracujícími na jednotlivých projektech. Každý člen projektového týmu bude mít předem definovanou funkci. Na základě těchto funkcí budou mít přiděleny pravomoci. Díky těmto pravomocím budou moci v IS editovat, mazat, aktualizovat. Jeden uživatel může pracovat na více projektech. Může mít tedy různé funkce a tedy i různé pravomoci.

#### 3.1.3 Kdo?

S IS budou pracovat uživatelé v následujících rolích:

- **Správce** – člověk starající se o IS
- **Manažer** – člověk, který má na starosti daný projekt, může vkládat nové členy týmu a rozděluje jim pravomoci
- **Členové týmu** – členové projektových týmů. Na základě přidělených funkcí (pravomocí) mohou editovat, mazat, aktualizovat

#### 3.1.4 Vstupy

U uživatele budeme evidovat login, jméno, příjmení, heslo, telefon, email.

U má právo budeme evidovat jednoznačné číslo práva, login, jednoznačné číslo funkce, jednoznačné číslo projektu.

U funkce budeme evidovat jednoznačné číslo funkce, název funkce, jednoznačné číslo pravomoci.

U projektu budeme evidovat jednoznačné číslo projektu, číslo smlouvy (popř. registračního čísla), název projektu, žadatele, datum zahájení, datum ukončení, výšku dotace, manažera projektu, popis projektu, partnera projektu, jeho cíl a stav.

U pravomoci budeme evidovat jednoznačné číslo pravomoci, editování, mazání, aktualizování.



U dokumentu budeme evidovat jednoznačné číslo dokumentu, samotný dokument, datum, jednoznačné číslo projektu a kategorie, login, název dokumentu, jeho velikost a typ.

U příspěvku budeme evidovat jednoznačné číslo odesílatele, příjemce, předmět, datum poslá-  
ní, text příspěvku, stav, stav2 a jednoznačné číslo dok\_pošty.

U dokumentu pošty budeme evidovat jednoznačné číslo dokumentu pošty, název dokumentu, jednoznačné číslo příspěvku, obsah dokumentu, jeho název, typ a velikost.

U kategorie budeme evidovat jednoznačné číslo kategorie, její název a jednoznačné číslo pro-  
jektu.

Každý uživatel může mít N má\_právo podle toho na kolika projektech pracuje. Každé  
má\_právo musí mít právě jednoho uživatele.

K jednomu má\_právo patří jedna funkce, stejnou funkci může mít více má\_právo.

Funkce musí mít nějaký typ pravomoci, stejné funkce mohou mít různé pravomoci.

Má\_právo je pro jeden projekt, projekt může mít N má\_právo.

Projekt může mít N kategorie, kategorie patří k nějakému projektu.

Projekt a kategorie v rámci projektu může obsahovat N dokumentů, každý dokument patří  
k nějakému projektu, respektive kategorii.

Uživatel může posílat a taky samozřejmě přijímat příspěvky, každý příspěvek patří k nějaké-  
mu odesílateli a příjemci.

Příspěvek může mít nějakou přílohu, příloha patří k nějakému příspěvku.

### 3.1.5 Výstupy

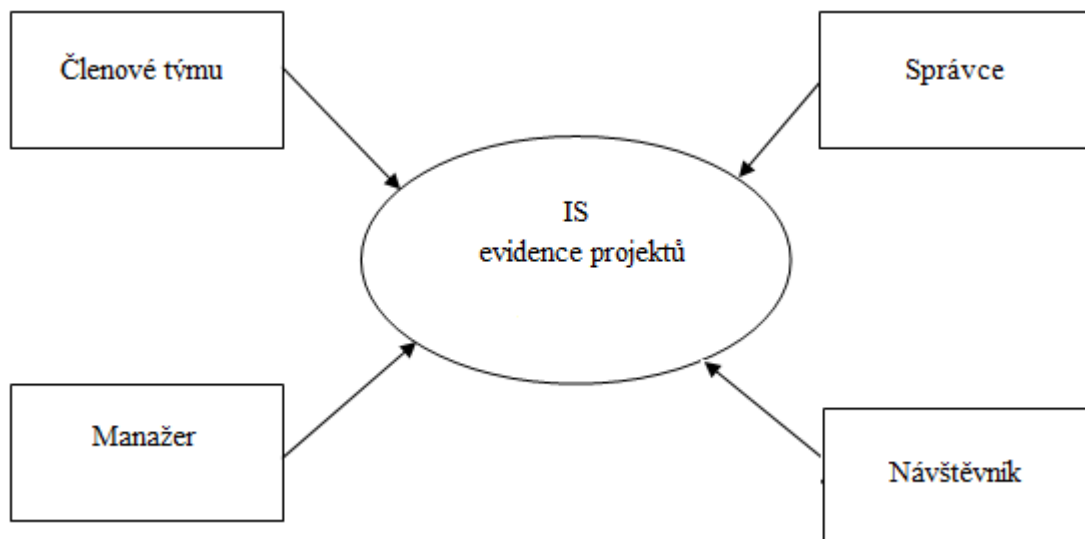
- **Seznam projektů a údajů o něm** – na úvodní stránce bude zobrazen seznam všech projektů
- **Seznam kategorií v rámci projektu** – v rámci každého projektu bude zobrazen seznam kate-  
gorií
- **Seznam členů týmu v rámci každého projektu** – v rámci některé z kategorií bude zobrazen  
seznam všech členů pracujících na daném projektu
- **Seznam dokumentů v rámci kategorie** – v rámci kategorie bude zobrazen seznam dokumen-  
tů

### 3.1.6 Vybrané funkce systému

Událost	Reakce	Aktér
Nový projekt, manažer	Zapiš do seznamu projektů, uživatelů	Správce
Editace projektu	Edituj daný projekt	Správce, Manažer
Smazání projektu	Po upozornění a potvrzení akce smaž data z odpovídajících tabulek podle daného id projektu	Správce
Změna stavu projektu	Nastav stav projektu	Správce, Manažer
Vložení člena týmu	Zapiš do seznamu uživatelů	Správce, Manažer
Editace člena týmu	Edituj daného člena	Správce, Manažer, Člen týmu
Smazání člena týmu	Po upozornění a potvrzení akce smaž data z odpovídajících tabulek podle daného loginu	Manažer, Správce
Vložení kategorie	Zapiš do seznamu kategorií	Správce, Manažer
Editace kategorie	Edituj danou kategorii	Správce, Manažer
Smazání kategorie	Po upozornění a potvrzení akce smaž data z odpovídajících tabulek podle daného id kategorie	Správce, Manažer
Vložení dokumentu	Zapiš do seznamu dokumentů	Správce, Manažer, Člen týmu
Smazání dokument	Smaž z tabulky Dokument	Správce, Manažer, Člen týmu
Odeslání pošty (s přílohou)	Zapiš do seznamu příspěvku, nastav stav i stav2 na nepřečteno a vlož dokument do tabulky Dok_pošty	Správce, Manažer, Člen týmu
Přečtení pošty	Nastav stav2 a stav příspěvku na přečteno.	Správce, Manažer, Člen týmu
Smazání příspěvku u uživatele	Nastav stav nebo stav2 příspěvku na smazáno.	Správce, Manažer, Člen týmu

Smazání příspěvku z databáze	Pokud je stav i stav2 nastaven na smazáno, smaž příspěvek z databáze	Správce, Manažer, Člen týmu
Výpis projektů	Zobraz seznam projektů	Správce, Manažer, Člen týmu, Návštěvník
Výpis kategorií v projektu	Zobraz seznam kategorií podle id projektu	Správce, Manažer, Člen týmu, Návštěvník
Výpis členů týmu projektu	Zobraz seznam uživatelů pracujících na projektu	Správce, Manažer, Člen týmu, Návštěvník
Výpis dokumentů	Zobraz dokumentů v rámci kategorie	Správce, Manažer, Člen týmu, Návštěvník

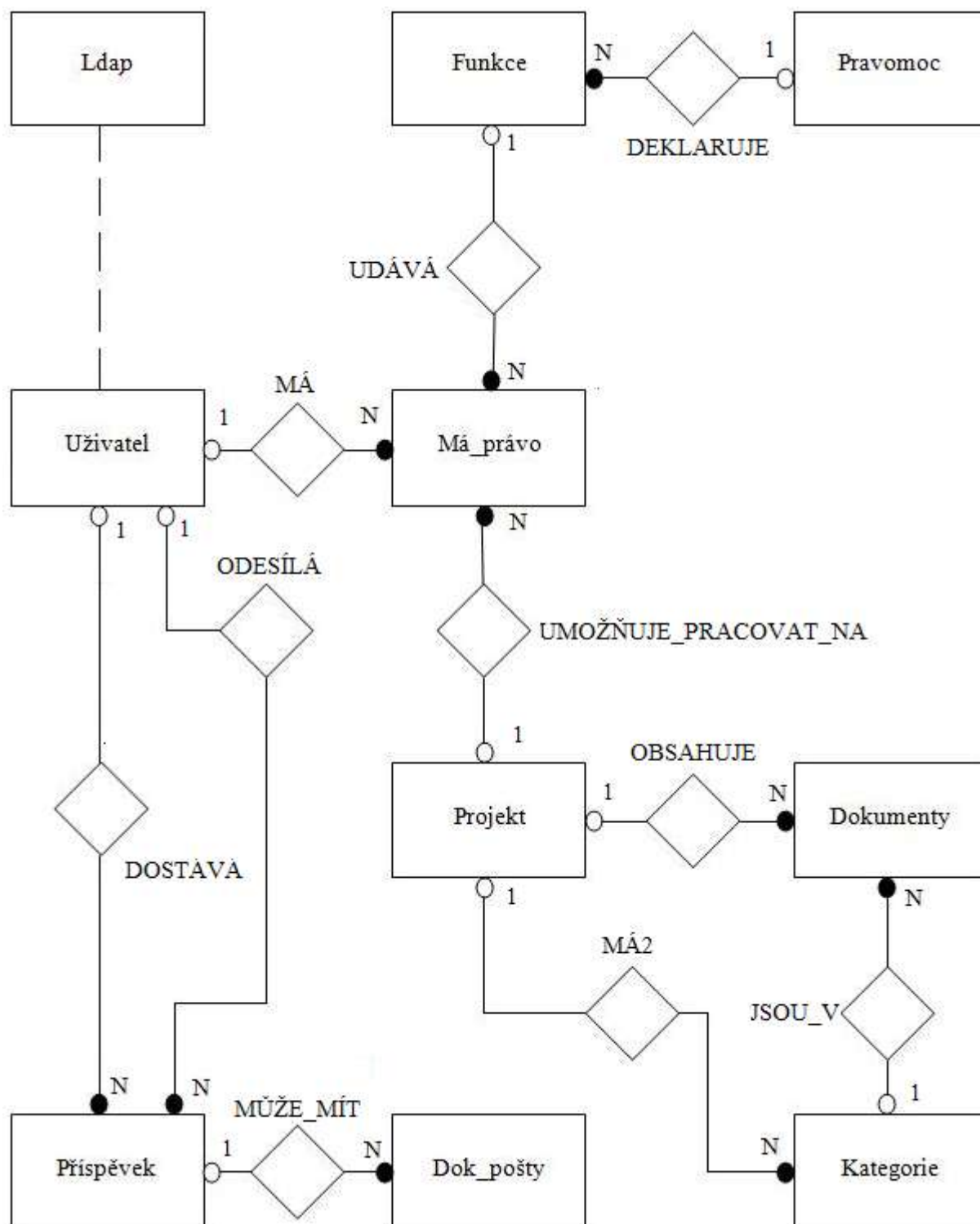
### 3.1.7 Okolí systému



Obrázek 37 – IS projekty - okolí systému

## 3.2 Datová analýza

### 3.2.1 E-R diagram



Obrázek 38 - IS projekty - E-R diagram

### 3.2.2 Lineární zápis typů entit

**Primární klíč, cizí klíč**

Uživatel (**login**, jméno, příjmení, heslo, telefon, email)

Má\_právo (**id\_má\_právo**, login, id\_projektu, id\_funkce)

Projekt (**id\_projektu**, číslo\_smlouvy, název\_projektu, žadatel, dat\_zahájení, podpora, dat\_ukončení, manažer, popis, partner\_projektu, cíl\_projektu, stav)

Funkce (**id\_funkce**, název\_funkce, id\_pravomoci)

Pravomoc (**id\_pravomoci**, editace, mazání, aktualizace)

Dokumenty (**id\_dokumentu**, dokument, datum, id\_projektu, id\_kategorie, login, název, typ, velikosti)

Příspěvek (**id\_příspěvku**, odesílatel, příjemce, předmět, datum, text, stav, stav2)

Příjemce (**id\_příjemce**, login, id\_příspěvku, přečetl)

Dok\_pošty (**id\_dok\_pošty**, id\_příspěvku, dokument, název, velikost, typ)

Kategorie (**id\_kategorie**, název\_kategorie, id\_projektu)

### 3.2.3 Lineární zápis typů vztahů

MÁ (Uživatel, Má\_právo)

UDÁVÁ (Má\_právo, Funkce)

UMOŽŇUJE\_PRACOVAT\_NA (Má\_právo, Projekt)

DEKLARUJE (Funkce, Pravomoc)

JE (Uživatel, Příjemce)

ODESÍLÁ (Uživatel, Příspěvek)

DOSTÁVÁ (Uživatel, Příspěvek)

MŮŽE\_MÍT (Příspěvek, Dok\_pošty)

OBSAHUJE (Projekt, Dokumenty)

MÁ2 (Projekt, Kategorie)

JSOU\_V (Dokument, Kategorie)

### 3.2.4 Datový slovník

#### Uživatel – seznam uživatelů, kteří mají přístup do systému

název	typ	vel.	klíč	null	index	Popis
login	char	6	PK	N	A	jednoznačná identifikace uživatele
jméno	varchar	15	N	N	N	jméno uživatele
příjmení	varchar	30	N	N	A	příjmení uživatele
heslo	char	20	N	N	N	heslo uživatele
telefon	char	13	N	N	N	telefon ve tvaru +429999999999
email	varchar	40	N	N	N	email uživatele

#### Projekt – seznam projektů

název	typ	vel.	klíč	null	index	Popis
id_projektu	int	10	PK	N	A	jednoznačná identifikace projektu
číslo_smlouvy	vachar	30	N	N	A	garantové číslo smlouvy nebo registrační číslo projektu
název_projektu	vachar	100	N	N	A	název projektu
žadatel	vachar	50	N	N	N	žadatel projektu
dat_zahájení	date		N	N	N	datum zahájení projektu
dat_ukončení	date		N	N	N	datum ukončení projektu
podpora	vachar	300	N	N	N	peněžní podpora projektu
manažer	vachar	50	N	N	A	jednoznačná identifikace, cizí klíč z tabulky uživatel
popis	vachar	1000	N	N	N	popis projektu
partner_projektu	vachar	100	N	A	N	kdo se ještě podílí na projektu
cíl_projektu	vachar	1000	N	N	N	cíle projektu
stav	vachar	10	N	N	N	stav projektu může nabývat hodnot v přípravě, ukončený úspěšně, probíhající apod.

**Má\_právo – tabulka, která určuje právo uživatelů k projektům**

název	typ	vel.	klíč	null	index	popis
id_má_právo	int	10	PK	N	A	jednoznačná identifikace má_právo
login	char	6	FK	N	A	jednoznačná identifikace uživatele, cizí klíč z tabulky Uživatel
id_projektu	int	10	FK	N	A	jednoznačná identifikace projektu, cizí klíč z tabulky Projekt
id_funkce	int	10	FK	N	A	jednoznačná identifikace funkce, cizí klíč z tabulky Funkce

**Funkce – seznam funkcí, které mohou mít uživatelé**

název	typ	vel.	klíč	null	index	popis
id_funkce	int	10	PK	N	A	jednoznačná identifikace funkce
název_funkce	vachar	50	N	N	A	název funkce
id_pravomoci	int	10	FK	N	A	jednoznačná identifikace pravomoci, cizí klíč z tabulky Pravomoc

**Pravomoc – seznam pravomocí, které mohou mít funkce**

název	typ	vel.	klíč	null	index	popis
id_pravomoci	int	10	PK	N	A	jednoznačná identifikace pravomoci
editace	bool		N	N	N	
mazání	bool		N	N	N	
aktualizace	bool		N	N	N	

**Kategorie – seznam kategorií jednotlivých projektů**

název	typ	vel.	klíč	null	index	popis
id_kategorie	int	10	PK	N	A	jednoznačná identifikace dokumentu
název_kategorie	vachar	200	N	N	N	název kategorie
id_projektu	int	10	N	N	A	jednoznačná identifikace projektu, cizí klíč z tabulky Projekt

### Dokumenty – seznam dokumentů, které vkládají uživatelé do kategorií

<b>název</b>	<b>typ</b>	<b>vel.</b>	<b>klíč</b>	<b>null</b>	<b>index</b>	<b>popis</b>
id_dokumentu	int	10	PK	N	A	jednoznačná identifikace dokumentu
dokument	blob		N	N	N	obsah dokument
datum	date		N	N	N	datum vložení
id_projektu	int	10	FK	N	A	jednoznačná identifikace projektu, cizí klíč z tabulky Projekt
id_kategorie	int	10	FK	N	A	jednoznačná identifikace projektu, cizí klíč z tabulky Kategorie
login	char	6	FK	N	A	jednoznačná identifikace uživatele, cizí klíč z tabulky Uživatel
název	vachar	45	N	N	A	název dokumentu
typ	vachar	45	N	N	N	typ dokumentu
velikost	int	10	N	N	N	velikost dokumentu

### Příspěvek – seznam příspěvků všech uživatelů

<b>název</b>	<b>typ</b>	<b>vel.</b>	<b>klíč</b>	<b>null</b>	<b>index</b>	<b>popis</b>
id_příspěvku	int	10	PK	N	A	jednoznačná identifikace příspěvku
odesílatel	char	6	FK	N	A	jednoznačná identifikace odesílatele, cizí klíč z tabulky Uživatel
příjemce	char	6	FK	N	A	jednoznačná identifikace příjemce, cizí klíč z tabulky Uživatel
předmět	vachar	50	N	N	N	předmět příspěvku
datum	date		N	N	N	datum odeslání příspěvku
text	vachar	2000	N	N	N	text příspěvku
stav	vachar	10	N	N	N	stav u odesílatele, přečtena, nepřečtena, smazána
stav2	vachar	10	N	N	N	stav u příjemce, přečtena, nepřečtena, smazána



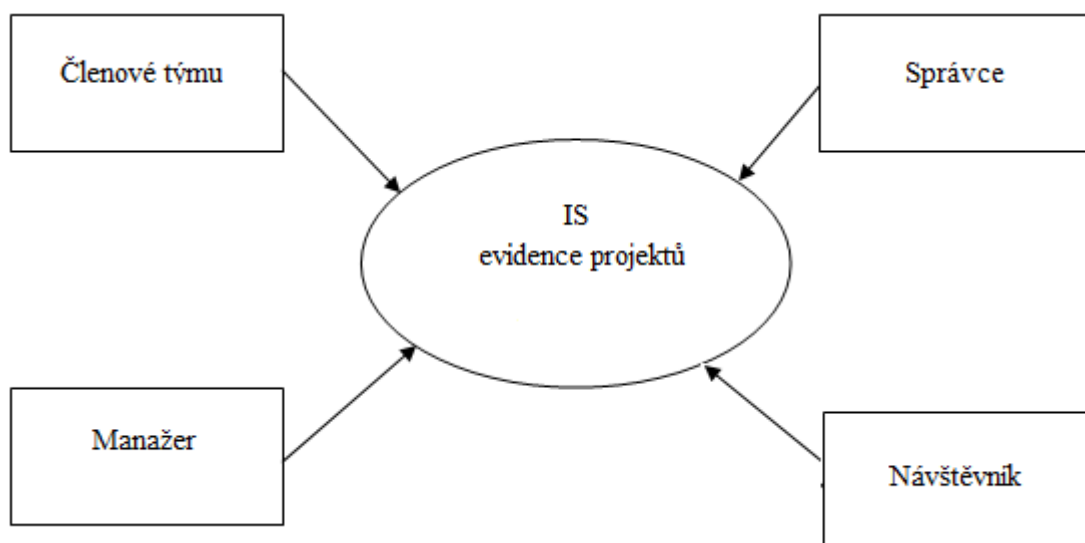
### Dok\_pošty – seznam dokumentů poslaný spolu s příspěvky

název	typ	vel.	klíč	null	index	popis
id_dok_pošty	int	10	PK	N	A	jednoznačná identifikace dok_pošty
id_příspěvku	int	10	FK	N	A	jednoznačná identifikace odesílatele, cizí klíč z tabulky Příspěvek
dokument	blob		N	N	N	obsah dokumentu
název	varchar	45	N	N	N	název dokumentu
typ	varchar	45	N	N	N	typ dokumentu
velikost	int	10	N	N	N	velikost dokumentu

Více podrobností o tvorbě datové analýzy lze nalézt v kapitole 2.3 nebo v [11].

## 3.3 Funkční analýza

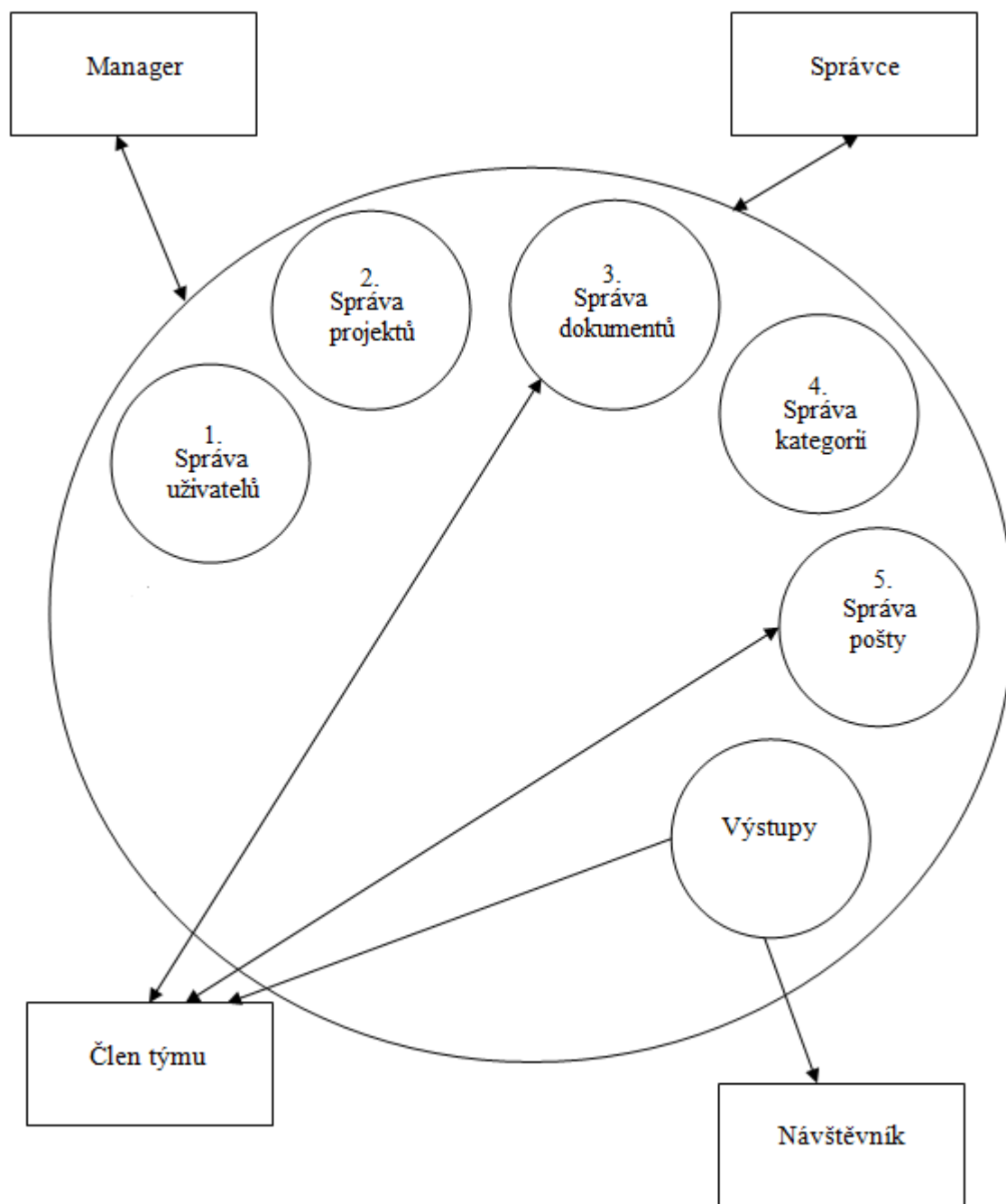
### 3.3.1 Kontextový diagram



Obrázek 39 - IS projekty - kontextový diagram

### 3.3.2 DFD

#### 3.3.2.1 DFD 0. úrovně

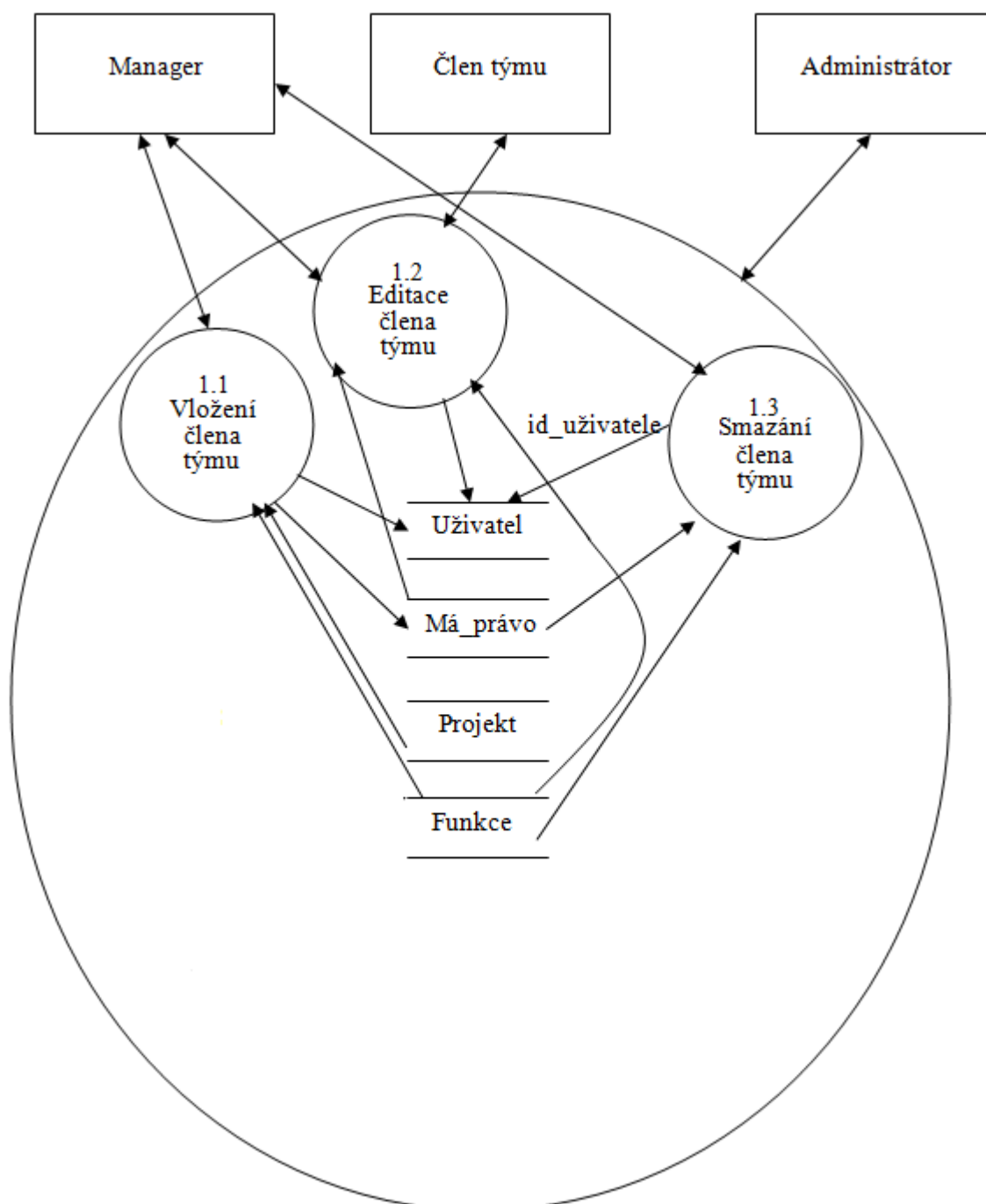


Obrázek 40 - IS projekty - DFD 0. Úrovně

V DFD 0. úrovně je možno vidět rozdělení IS do modulů jako správa uživatelů, projektů apod. Ty pak detailněji popisují tok dat nad daným modulem. Dále pak je možno vidět, kdo s jakým modulem smí pracovat.

### 3.3.2.2 DFD 1. Úrovně

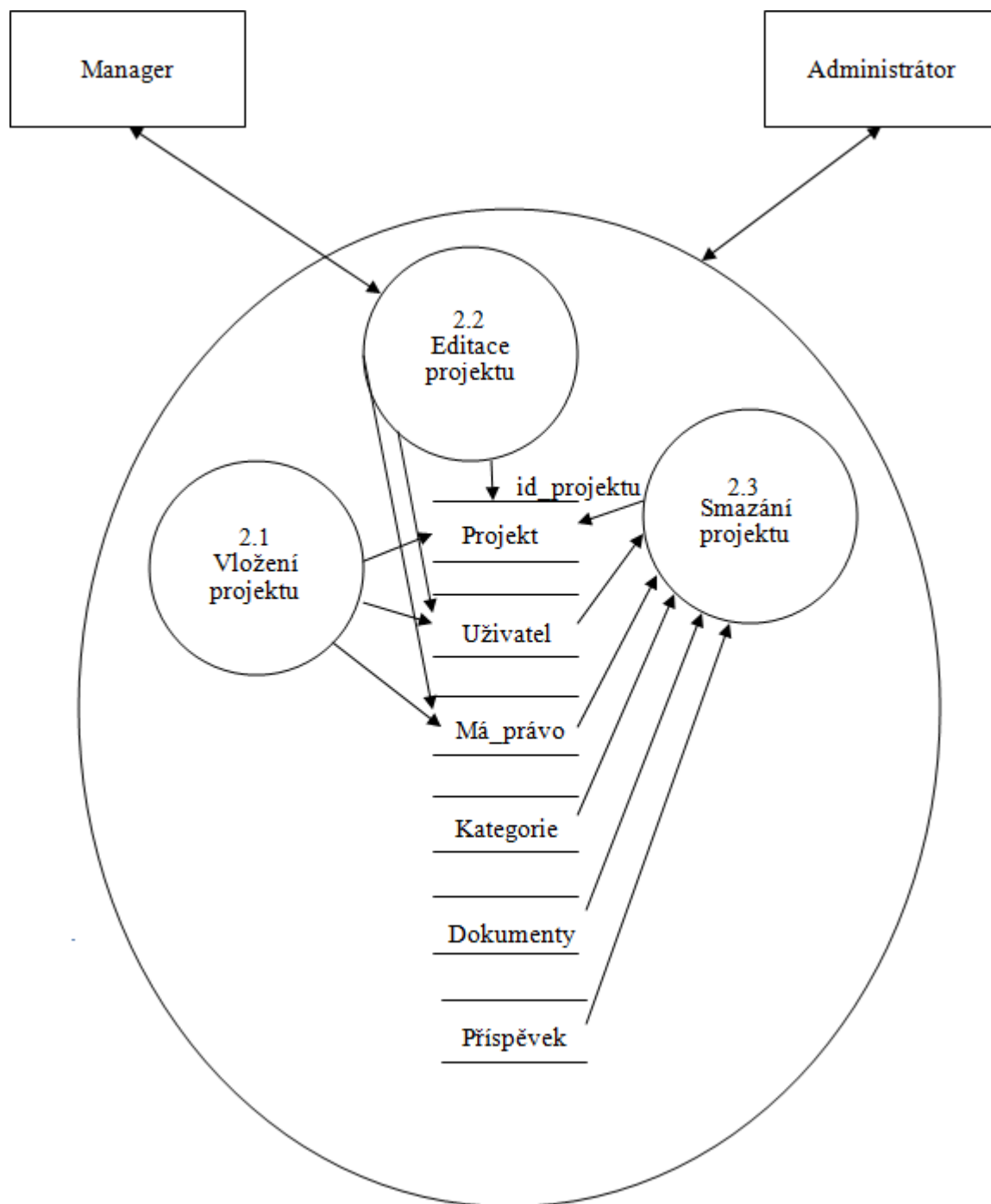
#### Správa uživatelů



Obrázek 41 - IS projekty - DFD 1. úrovně - správa uživatelů

DFD 1. Úrovně – Správa uživatelů zobrazuje akce vkládání, editace a mazání člena týmu. Vkládat a mazat člena týmu smí buď administrátor nebo manager. Editovat člena týmu, respektive údaje o sobě může každý člen týmu. Editovat funkci člena týmu smí pouze manager nebo správce.

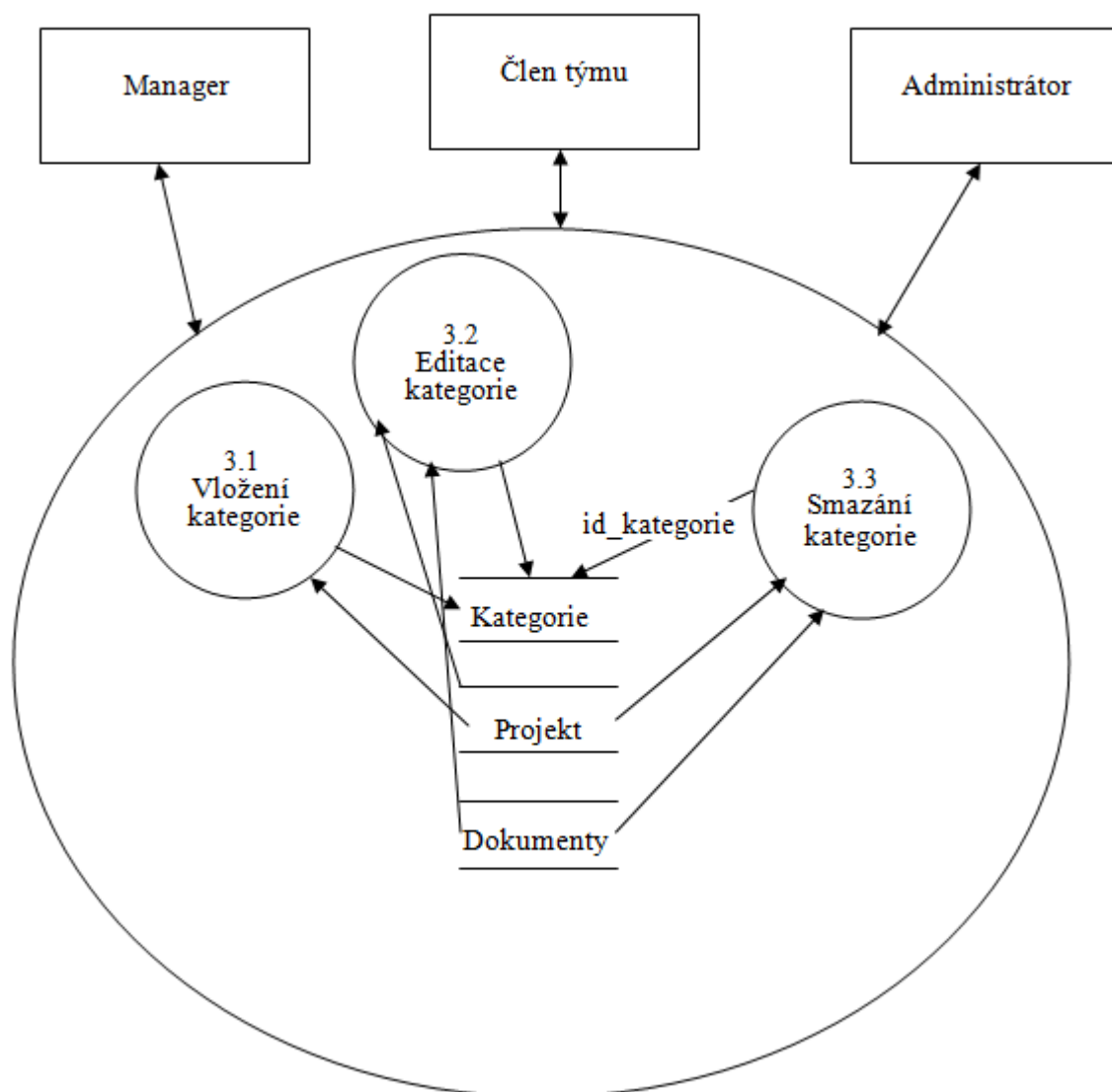
## Správa projektů



Obrázek 42 - IS projekty - DFD 1. úrovně - správa projektů

DFD 1. Úrovně – Správa projektů zobrazuje akce vkládání, editace a mazání projektů. Vkládat a mazat projekty smí administrátor. Editovat projekt pak i manager.

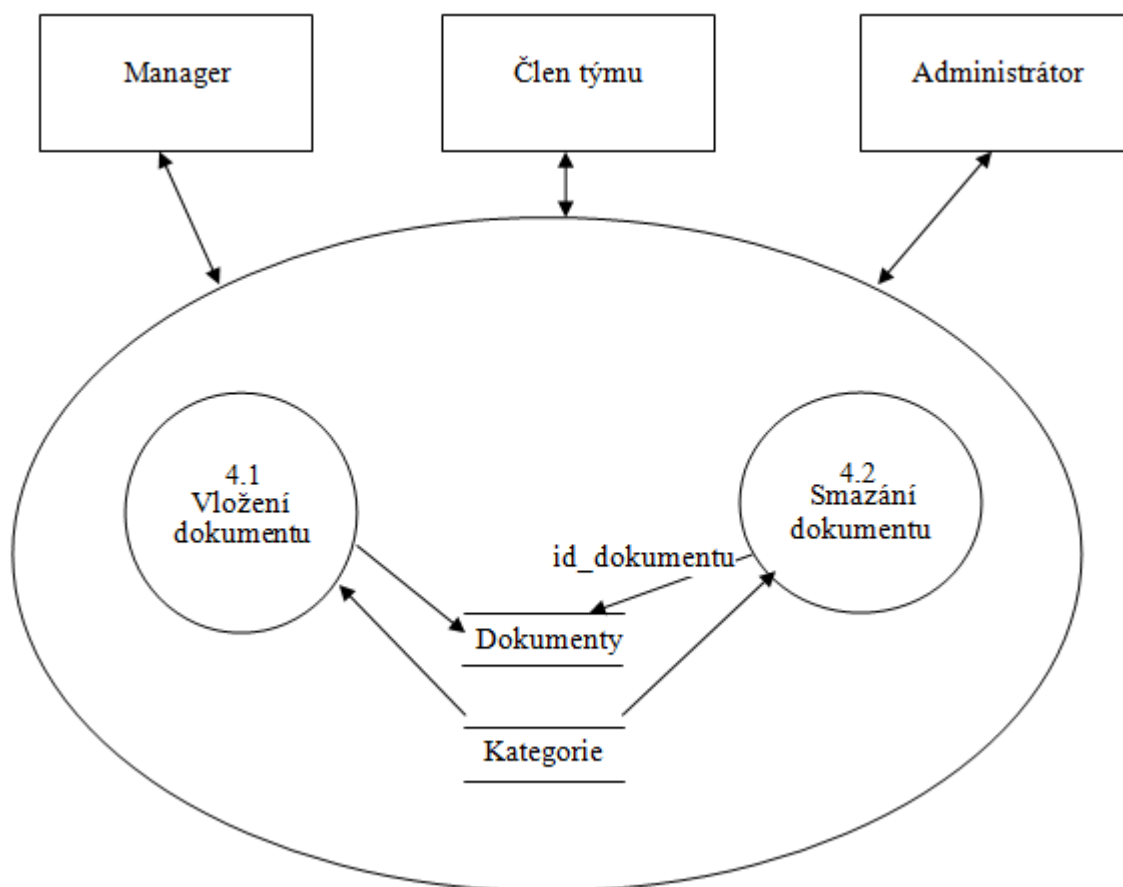
### Správa kategorií



Obrázek 43 - IS projekty – DFD 1. úrovně – správa kategorií

DFD 1. Úrovně – Správa kategorií zobrazuje akce vkládání, editace a mazání kategorií. Všechny tyto činnosti smí provádět osoba s příslušnými pravomocemi.

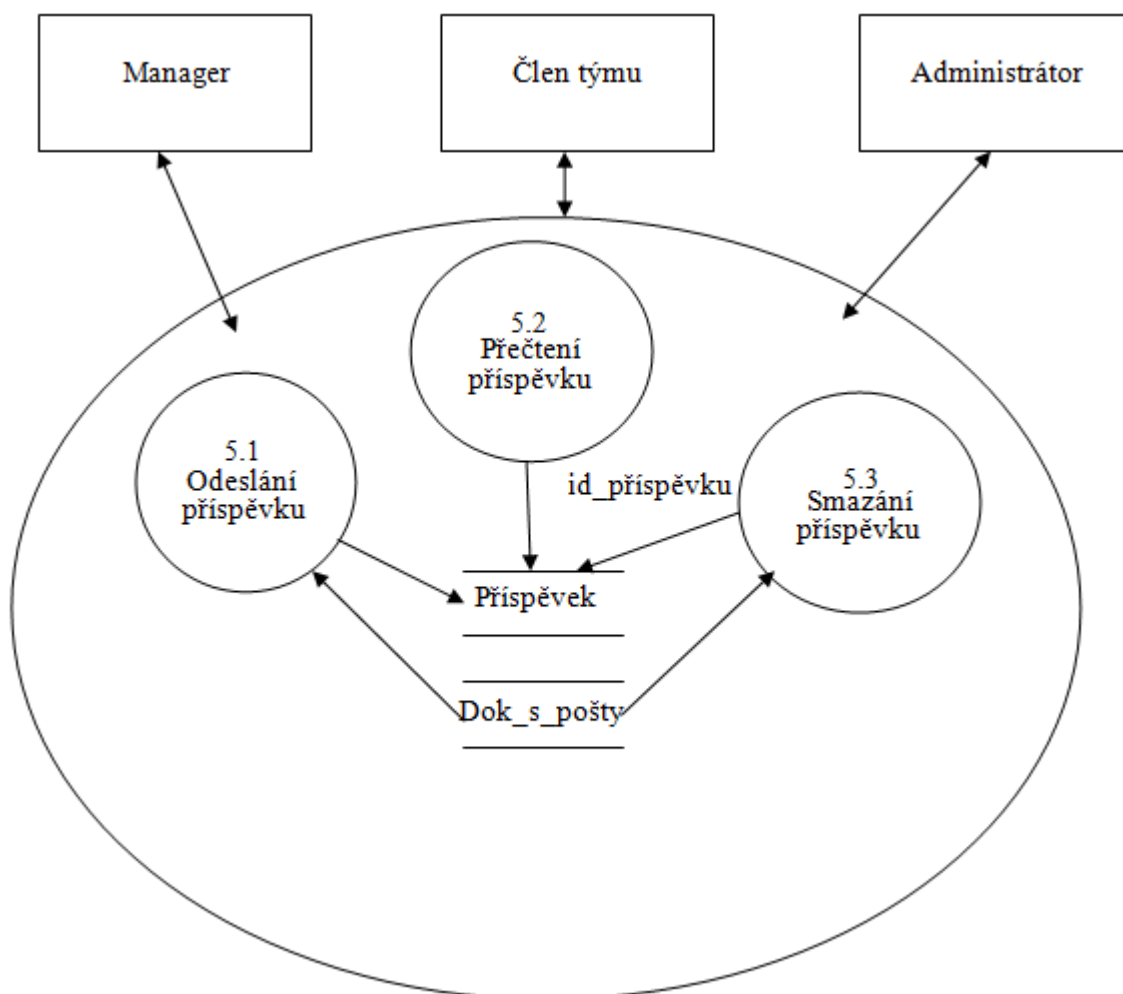
### Správa dokumentů



Obrázek 44 - IS projekty - DFD 1. úrovně správa dokumentů

DFD 1. Úrovně – Správa dokumentů zobrazuje akce vkládání a mazání dokumentů. Jako u předešlého DFD i zde mohou tyto činnosti provádět všichni členové týmu na základě svých pravomocí.

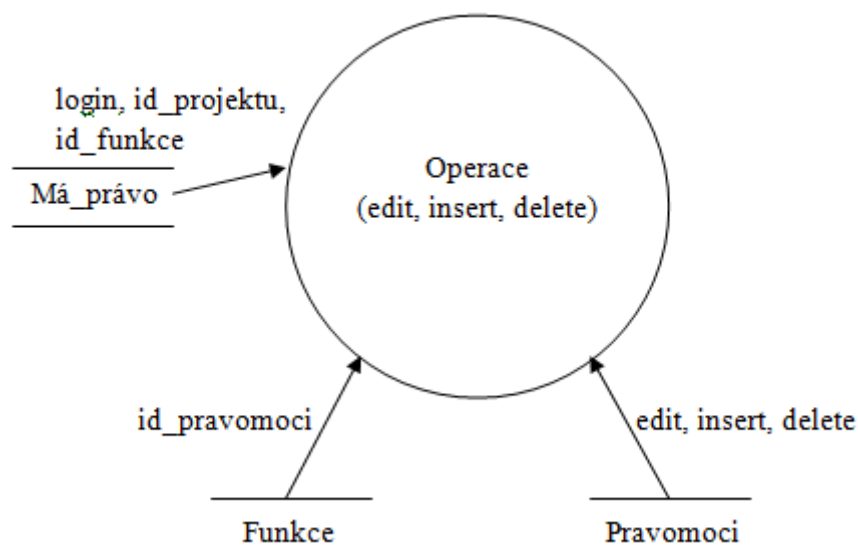
## Příspěvek



Obrázek 45 - IS projekty - DFD 1. úrovně – příspěvky

DFD 1. Úrovně – příspěvky zobrazuje akce odeslání, přečtení a smazání příspěvku. Tyto akce mohou provádět všichni ti, co mají přístup do systému.

## Obecné řešení přístupu k funkcím



Obrázek 46 - IS projekty - obecné řešení přístupu k funkcím

Při vytváření nového uživatele, je mu přidělena nějaká funkce, která má pravomoci vkládání (insert), editování (edit) a mazání (delete). Tyto pravomoci mohou pak nabývat hodnot true, jestliže je uživateli přidělena tato funkce nebo false.

### 3.3.3 Vybrané minispecifikace

#### 3.3.3.1 Vložení nového projektu (2.1)

1. Zobraz formulář pro zadání nového projektu s tlačítkem „vytvořit“.

Číslo smlouvy:	<input type="text"/>	Projekt:	<input type="text"/>
Manager(login):	<input type="text"/>	Žadatel:	<input type="text"/>
		Stav:	<input type="text"/>
Zahajení:	<input type="text"/>	Ukončení:	<input type="text"/>
Výše podpory:	<input type="text"/>	Partner projektu:	<input type="text"/>
Popis:	<input type="text"/>		
Cíl projektu:	<input type="text"/>		
<input type="button" value="vytvořit"/>			

Obrázek 47 - IS projekty - formulář pro vkládání projektu



2. Správce vyplní všechny položky formuláře a zmáčkne tlačítko „vytvořit“

3. JESTLIŽE formulář obsahuje chyby

PAK vypiš chybová hlášení a jdi na 2

JINAK

a) Vlož data z formuláře do databáze, do tabulky Projekt, vygeneruj nov0 id projektu a ulož ho do proměnné Pid\_projekt

b) Zkontroluj zda existuje uživatel s tímto loginem zadaný v poli Manager(login) v tabulce uživatel

JESTLIŽE uživatel s tímto loginem v tabulce není

PAK

a) získej podle zadaného loginu manager data z ldapu a ulož do pole Data

b) Vlož tyto získaná data do tabulky Uživatel jako novou větu s klíčem login



c) Vlož do tabulky Má\_právo, právo pro managera z formuláře na daný projekt. Funkce bude automaticky nastavena na managera a pravomoci na možnost editace, mazání a vkládání

d) Vyber z tabulky všechny jedinečné loginy z tabulky Má\_právo, kde id\_funkce = 1 (administrátor)

e) Vlož do tabulky Má\_právo, právo pro všechny loginy z předcházejícího bodu d), jejichž funkce = 1. Funkce bude automaticky nastavena na Administrátor a pravomoci na možnost editace, mazání a vkládání

### 3.3.3.2 Editování projektu (2.2)

1. Načti z databáze všechny řádky z tabulky Projekt do formuláře seznamu projektů s tlačítkem „Editovat“ a zobraz data ve formuláři pro vkládání projektu

Číslo smlouvy:	Název:	Stav:	Manager:	Začátek:	Konec:
Ax-093-oo98	Projekt 1 vpriprave	Petr Novák	2010-03-03	2012-03-03	 

Obrázek 48 - IS projekty - seznam projektů

2. Administrátor stiskne tlačítko „Editovat“ u projektu, který chce změnit.

3. a) Načti z databáze z tabulky Projekt

b) Zobraz data do polí formuláře s tlačítkem „Edit“ a zapamatuj si id\_projektu

Číslo smlouvy:	<input type="text"/>	Projekt:	<input type="text"/>
Manager(login):	<input type="text"/>	Žadatel:	<input type="text"/>
		Stav:	<input type="text"/>
Zahajeni:	<input type="text"/>	Ukončení:	<input type="text"/>
Výše podpory:	<input type="text"/>	Partner projektu:	<input type="text"/>
Popis:	<input type="text"/>		
Cíl projektu:	<input type="text"/>		
<input type="button" value="editovat"/>			

**Obrázek 49 - IS projekty - formulář pro editování projektu**

4. Administrátor provede změny a zmáčkne tlačítko „Editovat“
5. JESTLIŽE formulář obsahuje chyby  
PAK vypiš chybová hlášení a jdi na 2  
JINAK proved' update hodnot změnéné administrátorem pro příslušný řádek tabulky Projekt  
Pozn.  
JESTLIŽE administrátor změni managera projektu  
PAK proved' **3.3.4.2.1.**

### 3.3.4.2.1

- a) Vrat' počet řádku z tabulky Uživatel, kde login = Pmanager, z formuláře, z pole manager.
- b) JESTLIŽE je počet vrácených řádku = 0  
PAK získej data podle loginu managera z LDAPu a ulož do pole data.
- c) Vrat' z tabulky Má\_právo počet řádků, kde login = Pmanager a id projektu = Pid (id daného projektu)
- d) JESTLIŽE je počet řádku větší jak 0  
PAK proved' update daného řádku z tabulky Má\_právo, kde login = Pmanager a id projektu = Pid, id\_funkce nastav na 2 (funkce manager projektu)  
JINAK vlož nový řádek do tabulky Má\_právo, kde login = Pmanager, id projektu = Pid a id funkce = 2 (funkce manager projektu)
- e) Proved' update tabulky Projekt, nastav atribut managera na nový login managera podle id projektu
- f) Podle id projektu a loginu bývalého managera uloženého v proměnné Pmanager smaž řádek z tabulky Má\_právo
- g) Vrat' počet řádků z tabulky Má\_právo, podle id daného projektu, Pid a loginu bývalého managera uloženého v proměnné Pmanager
- h) JESTLIŽE je počet řádků = 0  
PAK Smaž řádek z tabulky Uživatel, kde login = Pbyv\_manager

### 3.3.3.3 Smazání projektu

1. Načti všechny řádky z tabulky Projekt do seznamu projektů s tlačítkem „Smazat“

Číslo smlouvy:	Název:	Stav:	Manager:	Začátek:	Konec:	
Ax-093-oo98	Projekt 1 vpriprave	Petr Novák	2010-03-03	2012-03-03	✖	✎

Obrázek 50 - IS projekty - seznam projektů

2. Administrátor stiskne tlačítko „Smazat“ u projektu, který chce smazat
3. Po potvrzení akce načti id projektu do proměnné Pid.
4. Smaž řádek z tabulky Projekts, kde id\_projektu = Pid

### 3.3.3.4 Vložení člena týmu

1. a) Zobraz formulář pro vložení nového člena týmu s tlačítkem „Vytvořit“  
b) Načti z tabulky z tabulky Projekt, všechny id projektu a názvy projektu do seznamu projektů  
c) Načti z tabulky Funkce do seznamu funkcí všechny id funkce a názvy funkcí

Projekt:	<input type="text"/>	Uživatel(login):	<input type="text"/>
Funkce:	<input type="text"/>	Nová funkce:	<input type="text"/>
Pravomoce:	Editování: <input type="checkbox"/>	Mazání: <input type="checkbox"/>	Vkládání: <input type="checkbox"/>
<input type="button" value="Vytvořit"/>			

Obrázek 51 - IS projekty - formulář pro vložení člena týmu

2. Manager nebo administrátor vyplní formulář a stiskne tlačítko „Vytvořit“
3. JESTLIŽE formulář obsahuje chyby  
PAK vypiš chybová hlášení a jdi na 2  
JINAK
  - a) Vlož do proměnné Pid id vybraného projektu
  - b) Vrať z databáze počet řádku z tabulky Má\_právo, kde login = login zadaný ve formuláři
  - c) JESTLIŽE je počet řádků roven 0  
PAK
    - a) podle loginu získej data z LDAPu a ulož do pole Data
    - b) JESTLIŽE je vybrána funkce ze seznamu  
PAK vrať z databáze řádek tabulky Uživatel, kde login = hodnota zadaná ve formuláři  
JESTLIŽE je počet řádku v databázi v tabulce Uživatel roven nule  
PAK vlož data z pole Data[] do databáze, do tabulky Uživatel

c) Vlož data do tabulky Má\_právo

JINAK vypiš hlášení „Uživatel s tímto loginem je již členem týmu v tomto projektu.  
Pokud toto chcete změnit, smažte ho a vytvořte znovu“

### 3.3.3.5 Odeslání příspěvku

1. a) Zobraz formulář pro odeslání nové pošty s tlačítkem „Odeslat“  
b) Načti z tabulky Projekt do seznamu projektů všechny id projektu a názvy projektů

Komu:  Všem lidem z projektu:  ▼

Příloha:



Předmět:

**Obrázek 52 - IS projekty - formulář pro odesílání příspěvku**

2. Uživatel vyplní položky a stiskne tlačítko „Odeslat“
3. JESTLIŽE formulář obsahuje chyby  
PAK vypiš chybová hlášení a jdi na 2  
JINAK
  - a) JESTLIŽE pole formuláře předmět je prázdné  
PAK proměnná Ppredmet = NULL  
JINAK proměnná Ppredmet = obsah formuláře pole předmět
  - b) Vlož hodnoty z polí formuláře a proměnnou Ppredmet do databáze do tabulky Příspěvek, vygeneruj id příspěvku a ulož ho do proměnné Pid\_příspěvku
  - c) JESTLIŽE není pole pro vložení příspěvku prázdné  
PAK vlož do tabulky Dok\_pošty spolu s proměnnou Pid\_příspěvku soubor odeslaný s příspěvkem

### 3.3.3.6 Smazání příspěvku

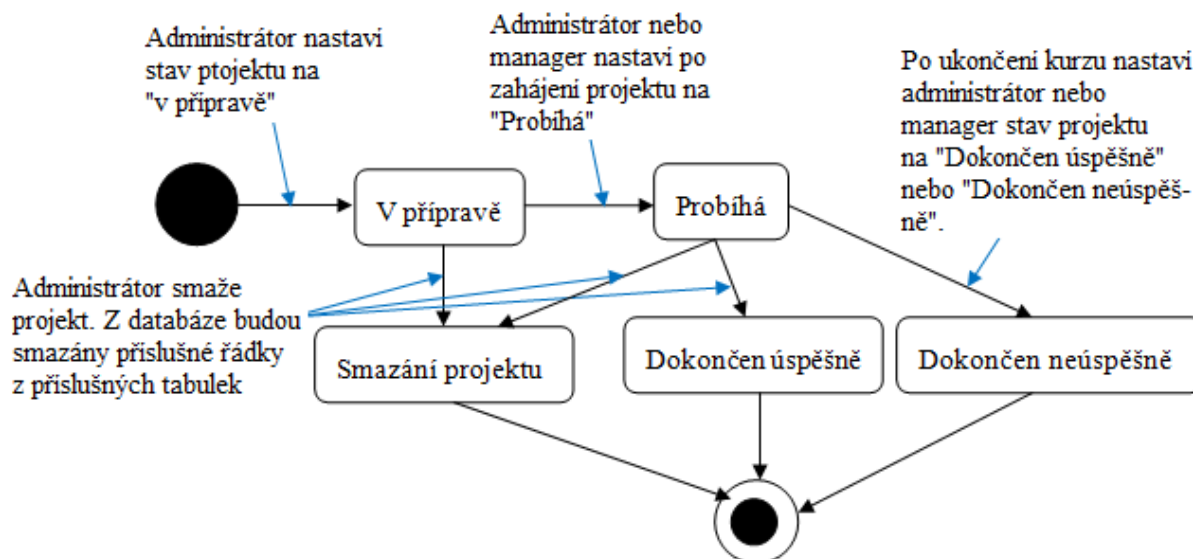
1. Načti z databáze všechny řádky z tabulky Příspěvek do seznamu přečtené pošty, kde příjemce je přihlášený uživatel s tlačítkem „Smazat“

Datum:	Odesílatel:	Předmět:	Akce:
2010-03-20	mec063	Zkouska	 

Obrázek 53 - IS projekty - seznam přečtené pošty

2. Uživatel vybere ze seznamu příspěvek, který chce smazat a stiskne tlačítko „Smazat“
3. Vlož do proměnné Pid id příspěvku, který chce uživatel smazat
4. Proveď update příspěvku, kde id příspěvku se rovná proměnné Pid a nastav hodnotu stav2 v databázi na „smazáno“
5. Vrať z databáze hodnotu atributu stav a stav2, kde id příspěvku se rovná proměnné Pid
6. JESTLIŽE se stav a stav2 rovná „smazáno“  
PAK smaž příspěvek z tabulky Příspěvek, kde id příspěvku se rovná proměnné Pid
7. Vrať z databáze hodnotu id\_dok\_pošty z tabulky Dok\_pošty, kde id příspěvku se rovná Pid
8. JESTLIŽE počet vrácených řádků větší jak 0  
PAK smaž řádek z tabulky Dok\_pošty, kde id pošty = Pid

### 3.3.4 Časová analýza stavu projektu



Obrázek 54 - Časová analýza stavu projektu

## 3.4 Návrh implementace

### 3.4.1 Použité technologie

Jako u předchozího projektu, tak i zde bylo k implementaci využito technologií **XHTML**, **PHP**, **MySQL** a **JAVASCRIPT**.

V rámci PHP byl využit protokol LDAP, a to z důvodu toho, že do systému mají možnost vstoupit zaměstnanci VŠB – TU Ostrava. Pomocí tohoto protokolu se získávají data o uživateli pracujících na jednotlivých projektech, tj. login, jméno, příjmení, email. Dále se pomocí LDAPu ověřují platnost hesel pro vstup do systému, jelikož pro vstup je nutné zadat právě heslo ze školního adresářového serveru ldap.vsb.cz.

#### 3.4.1.1 LDAP - obecně

Jedná se o definovaný protokol pro přístup a ukládání dat na adresářovém serveru nad TCP/IP. Podle tohoto protokolu jsou jednotlivé položky na serveru ukládány formou záznamů a uspořádány do stromové struktury. Je vhodný pro udržování adresářů a práci s informacemi o uživateli (např. pro vyhledávání adres konkrétních uživatelů v příslušných adresářích, resp. databázích).

#### LDAP se skládá ze čtyř modelů

- informační model
- jmenný model
- funkční model
- bezpečnostní model

**Informační model** - informace jsou v adresáři uloženy ve stromové struktuře (**Directory Information Tree**), jehož kořenem je **rootDS** obsahující globální informace o adresáři. Tento model obsahuje záznamy, které obsahují informace o objektu (př. student na VŠB – TUO). Tyto objekty se pak skládají z atributů, které mají vždy typ a nějakou hodnotu.

Implementace informačního modelu se označuje jako **schéma**. Toto schéma definuje třídy objektů a atributy.

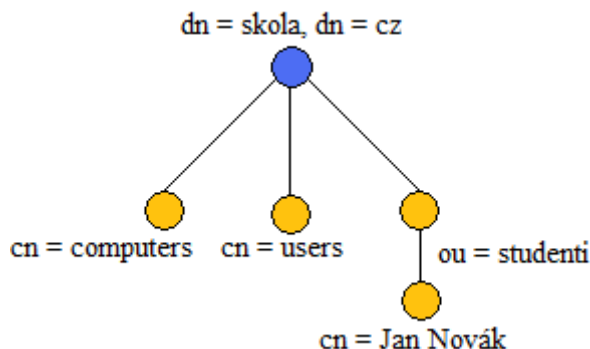
**Třídy objektů (objectClass)** jsou kategorie objektů, které mohou být vytvořeny v adresáři. Např. user, komputer apod.

**Atributy objektů (object attributes)** - vlastnosti objektů. Atribut může obsahovat jednu nebo více hodnot. Určité atributy patří k určité třídě objektů. Schéma určuje datové typy atributů a určuje, které hodnoty musí být vyplněny.

Podle polohy objektu ve stromové struktuře, se jedná buď o **list**, tzn. že nemá potomky nebo o **kontejner**, který může obsahovat jeden či více objektů.

**Jmenný model** – K jednoznačné identifikaci objektů se používá **Distinguished Name (DN)**. Tento identifikátor také obsahuje pozici objektu ve stromě a skládá se ze jména objektu a jmen jednotlivých kontejnerů a domén, které obsahují objekt. Ty jsou odděleny čárkou.

**Příklad části adresáře AD pro doménu skola.cz**



**Obrázek 55 - LDAP příklad**

V organizační jednotce (OU) studenti je umístěn uživatel Jan Novák, pro kterého je DN = cn=Jan Novák,ou=studenti,dc=skola,dc=cz.

**Funkční model** - definuje, co se může provádět s informacemi v adresáři. Jedná se o 9 operací, které jsou zařazeny do 3 funkčních oblastí. Např. v oblasti autentizace je to funkce **bind**, která inicializuje spojení, vyjednáva o metodě autentizace a provádí ji.

**Bezpečnostní model** – tento model určuje, jak se přistupuje k datům po bezpečnostní stránce. Souvisí s autentizačními službami z adresářových služeb.

Více podrobností zde [3], [4], [15].

**Ukázka zdrojového kódu IS:**

**Funkce, pomocí které získávám data o uživateli:**

```

function getDataFromLdap($login)
{
    define('ACCESS_LDAP_SERVER', 'ldap.vsb.cz'); // Definice konstanty ACCESS_LDAP_SERVER
    $ds = ldap_connect(ACCESS_LDAP_SERVER);
    // Funkce ldap vytvoří spojení na zadaný LDAP server a uloží do proměnné
    // $ds, funkce vrátí id linky nebo false
    if(!$ds) {
        error_label('cannot_connect_ldap_server', ERROR_CANNOT_CONNECT_LDAP_SERVER, 'yes');
    } else {
        $r = ldap_bind($ds); // Navázání k adresáři pomocí ID linky, anonymní bind
        $sr = ldap_search($ds, "", "cn=$login"); // Pomocí této funkce získáme id výsledku hledání
        // podle zadaného loginu
        $info = ldap_get_entries($ds, $sr); // funkce, pomocí níž získáme záznamy vrácené funkcí
        // ldap_search jako vícerozměrné pole
        if ($info["count"] == 1) { // Jestliže nám předchozí funkce vrátí výsledek, pak vytvoř
            // pole data a vlož do nich údaje o uživateli
            $data = array();
            $u_dn = $info[0]["dn"]; // celé "domain name" (cn=X,ou=Y,o=Z)
            $u_mail = $info[0]["mail"][0];
        }
    }
}
  
```

```

    .....
}
}
return $data; // vrať pole data
}

```

### Funkce, pomocí které ověřuji heslo na lap severu:

```

function ldap_verification($login,$password_ldap)
{
    define('ACCESS_LDAP_SERVER', 'ldap.vsb.cz'); // Definice konstanty ACCES_LDAP_SERVER
    $ds = ldap_connect(ACCESS_LDAP_SERVER); //připojení k LDAP serveru
    if (!$ds) {
        error_label('cannot_connect_ldap_server',ERROR_CANNOT_CONNECT_LDAP_SERVER,'yes');
    } else {
        $r = ldap_bind($ds); // Navázání k adresáři pomocí ID linky, anonymní bind
        $sr = ldap_search($ds,"","cn=$login"); // Pomocí této funkce získáme id výsledku hledání
        // podle zadaného loginu
        $info = ldap_get_entries($ds, $sr); // funkce, pomocí níž získáme záznamy vrácené funkcí
        // ldap_search jako vícerozměrné pole

        if ($info["count"] == 1) {
            $u_dn = $info[0]["dn"]; // celé "domain name" (cn=X,ou=Y,o=Z)
            $u_login=""; $u_fakulta="";
            $u_adn = split(",",$u_dn,4); // pole retezcu {"NA ZEV = HODNOTA", ... }
            for($i=0; $i<count($u_adn); $i++){
                $u_aitem = split("=", $u_adn[$i],2); // pole {"NAZEV", "HODNO TA"}
                if(count($u_aitem) == 2){
                    if($u_aitem[0]=="cn") $u_login = $u_aitem[1]; //do proměnné $u_login vložím
                    // login, který dostaneme z ldapu
                    if($u_aitem[0]=="o") $u_fakulta = $u_aitem[1];
                }
            }
            $ds1 = ldap_connect(ACCESS_LDAP_SERVER); // vytvoříme nové při pojení k lap serveru
            if (empty($u_login) || empty($password_ldap) || !@ldap_bind($ds1, $u_dn,
            $password_ldap)) {
                // Jestliže nedostanu vrácen login nebo nezadáme heslo nebo se neprovede bind,tak se ne-
                provede žádná další akce
            } else { //jinak uzavři spojení, proved dané akce
                ldap_close($ds1);
                ldap_close($ds);
                header ("Locati on:http://homel.vsb.cz/~mec063/projekty/sys/main.php");
                .....
                return true; // vrať true
            }
            ldap_close($ds1); // uzavři spojení
            ldap_close($ds); // uzavři spojení
        }
        return false; // vrať false
    }
}

```

## 3.4.2 Hardware, Software

### Viz. Kapitola 2.4.3.



### 3.4.3 Změna v datových slovnících

U tabulek **Dokumenty** a **Dok\_pošty**:

Atribut **dokument** změněn z datového typu **blob** na datový typ **mediumblob** z důvodu toho, že původní datový typ je z hlediska jeho velikosti nedostačující. Maximální velikost datového typu **blob** je 65535 bytů a velikost datového typu **mediumblob** je 16777215.

U tabulky **Pravomoc**:

Atributy vkládání, mazání a editace změněny z datového typu **Boolean**, který nabývá hodnot true nebo false na datový typ **TINYINT**, což je celočíselný datový typ, který může nabývat hodnot od -128 do 127. My využijeme hodnotu 0, která nám nahradí původní hodnotu false a hodnotu 1, která nám nahradí původní hodnotu true. Datový typ byl změněn z důvodu toho, že datový typ **Boolean** se v MySQL databázi nevyskytuje.

### 3.4.4 Indexová analýza

Jako v předchozím příkladě, tak i zde jsou vytvořeny indexy pro všechny primární klíče, dále pak pro atributy, které spojují tabulky, tj. cizí klíče a pro atributy, podle kterých jsou tříděny různé seznamy nebo využito vyhledávání. U všech těchto atributů byly použity udržované indexy, jelikož jsou neustále používány.

**Uživatel**

název	typ	vel.	klíč	null	index	Popis
login	char	6	PK	N	A	jednoznačná identifikace uživatele
příjmení	varchar	30	N	N	A	příjmení uživatele

**Kategorie**

id_kategorie	int	10	PK	N	A	jednoznačná identifikace dokumentu
id_projektu	int	10	N	N	A	jednoznačná identifikace projektu, cizí klíč z tabulky Projekt

**Projekt**

id_projektu	int	10	PK	N	A	jednoznačná identifikace projektu
číslo_smlouvy	vachar	30	N	N	A	garantové číslo smlouvy nebo registrační číslo projektu
název_projektu	vachar	100	N	N	A	název projektu
manažer	vachar	50	N	N	A	jednoznačná identifikace manažera, cizí klíč z tabulky uživatel

### Má\_právo

id_má_právo	int	10	PK	N	A	jednoznačná identifikace má_právo
login	char	6	FK	N	A	jednoznačná identifikace uživatele, cizí klíč z tabulky Uživatel
id_projektu	int	10	FK	N	A	jednoznačná identifikace projektu, cizí klíč z tabulky Projekt
id_funkce	int	10	FK	N	A	jednoznačná identifikace funkce, cizí klíč z tabulky Funkce

### Funkce

id_funkce	int	10	PK	N	udržovaný	jednoznačná identifikace funkce
název_funkce	vachar	50	N	N	udržovaný	název funkce
id_pravomoci	int	10	FK	N	udržovaný	jednoznačná identifikace pravomoci, cizí klíč z tabulky Pravomoc

### Dokumenty

id_dokumentu	int	10	PK	N	A	jednoznačná identifikace dokumentu
id_projektu	int	10	FK	N	A	jednoznačná identifikace projektu, cizí klíč z tabulky Projekt
id_kategorie	int	10	FK	N	A	jednoznačná identifikace projektu, cizí klíč z tabulky Kategorie
login	char	6	FK	N	A	jednoznačná identifikace uživatele, cizí klíč z tabulky Uživatel
název	vachar	45	N	N	A	název dokumentu

### Příspěvek

id_příspěvku	int	10	PK	N	A	jednoznačná identifikace příspěvku
odesílatel	char	6	FK	N	A	jednoznačná identifikace odesílatele, cizí klíč z tabulky Uživatel
příjemce	char	6	FK	N	A	jednoznačná identifikace příjemce, cizí klíč z tabulky Uživatel

### Pravomoc

id_pravomoci	int	10	PK	N	A	jednoznačná identifikace pravomoci
--------------	-----	----	----	---	---	------------------------------------

## Dok\_pošty

id_dok_pošty	int	10	PK	N	A	jednoznačná identifikace dok_pošty
id_příspěvku	int	10	FK	N	A	jednoznačná identifikace odesílatele, cizí klíč z tabulky Příspěvek

## 3.5 Popis Implementace

### 3.5.1 Uživatelská příručka

V této kapitole budou popsány některé funkce IS jako přihlášení, jednotlivé sekce IS atd.

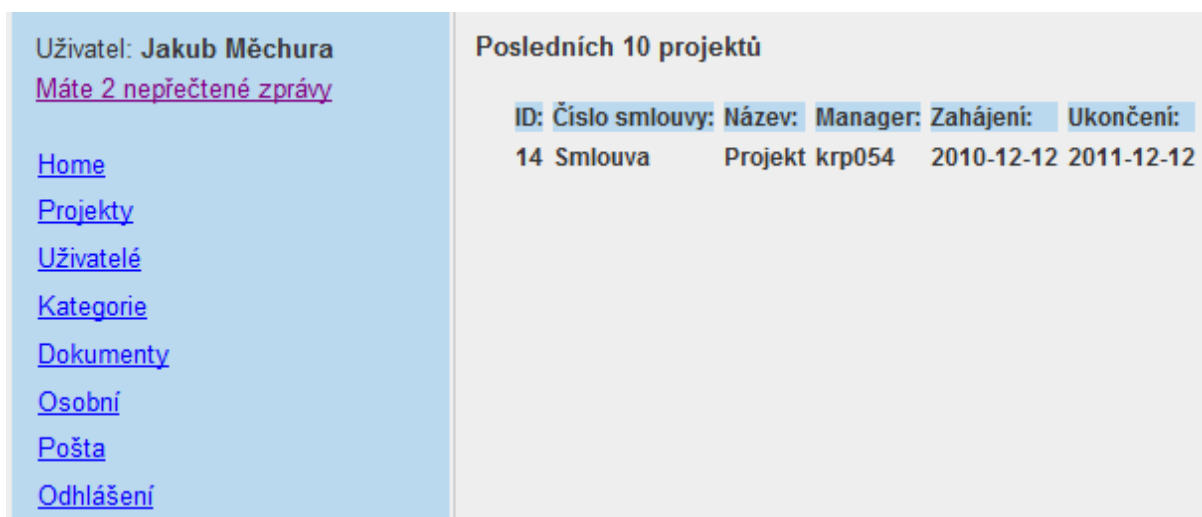
#### 3.5.1.1 Přihlášení



Obrázek 56 - IS projekty - přihlašování do systému

1. Uživatel zadá svůj login a heslo z LDAPu a stiskne tlačítko „Přihlásit“.

#### 3.5.1.2 IS – úvodní obrazovka



ID:	Číslo smlouvy:	Název:	Manager:	Zahájení:	Ukončení:
14	Smlouva	Projekt krp054		2010-12-12	2011-12-12

Obrázek 57 - IS projekty - úvodní obrazovka

### 3.5.1.3 IS - sekce Projekty

Projekty

Číslo smlouvy:

Projekt:

Manager(login):

Žadatel:

Stav:

Zahájení:

Ukončení:

Výše podpory:

Partner projektu:

Popis:

Číslo projektu:

editovat

vytvořit

Číslo smlouvy:

Název:

Stav:

Manager:

Začátek:

Konec:

00/99-sa

Projekt 2

vpriprave

Tomas Mollnhuber

2010-12-12

2014-12-12

✖

✎

Obrázek 58 - IS projekty - sekce projekty

1. Správce má možnost editovat, mazat a vkládat projekt.
2. V případě, že chce vložit nový (např. nového správce) vyplní všechny položky a stiskne tlačítko „Vytvořit“.
3. V případě, že bude chtít editovat údaje o projektu, vybere příslušný projekt a zmáčkne tlačítko se symbolem tužky. Změní údaje a zmáčkne tlačítko „editovat“.
4. Jestliže bude chtít správce smazat projekt, učiní tak stisknutím tlačítka se symbolem křížku. S projektem budou smazány všichni členové týmu, kategorie, dokumenty atd.
5. Současně s projektem je vložen i manager projektu, který má právo editovat projekt.
6. Ostatní členové projektu vidí pouze seznam, který lze po kliknutí na název sloupce abecedně seřadit podle daného sloupce.

#### 3.5.1.4 IS – sekce Uživatelé

**Uživatelé**

Projekt:  Uživatel(login):   
Funkce:  Nová funkce:   
Pravomoc: Editování: ☐ Mazání: ☐ Vkládání: ☐

Projekt:	Funkce:	Login:	Příjmení:	Jméno:	Email:
Projekt 1	Administrátor	mec063	Měchura	Jakub	mec063@vsb.cz

Obrázek 59 - IS projekty - sekce uživatelé

1. Uživatelé (členy týmu) může vkládat správce nebo manager daného projektu.
2. Správce nebo manager vyplní formulář a stiskne tlačítko vytvořit.
3. Vyplnit funkci člena týmu lze dvěma způsoby a to:
  - a) Buď vybere se seznamu již existujících funkcí.
  - b) Vytvoří novou funkci a zatrhne pravomoci.
4. Manager a správce může editovat (měnit funkce a pravomoci) členů týmu, kteří spadají pod daného managera, poté stiskne tlačítko „editovat“.

#### 3.5.1.5 IS – sekce Kategorie

**Kategorie**

Projekt:  Kategorie:

Projekt 1x  
Projekt 2  
Projekt34  
vdfsasa  
Projekt 1  
Projekt 1x

Obrázek 60 - IS projekty - sekce kategorie před výběráním projektu

1. V seznamu projektů jsou projekty, v kterých figuruje přihlášený uživatel jako člen týmu. Po vybrání některého z projektů se mu zobrazí tlačítka editovat a vytvořit podle toho, jaké má nastavená práva. U všech projektů jsou automaticky vytvořeny kategorie Úvod, Členové týmu a Dokumenty.

Projekt:  Kategorie:

Obrázek 61 - IS projekty - sekce kategorie po vybrání projektu

2. Pokud chce uživatel vytvořit novou kategorii, vybere projekt, zadá kategorii a stiskne tlačítko „vytvořit“.
3. Pokud chce uživatel editovat kategorii, vybere ze seznamu kategorií, stiskne symbol tužky, provede změny a stiskne tlačítko „editovat“.

**Projekt:** **Kategorie:**

Projekt 1x Kategorie 1  

Projekt 2 Kategorie 1  

Obrázek 62 - IS projekty - seznam kategorií


4. Pro smazání slouží symbol ve tvaru tužky.

### 3.5.1.6 IS – sekce Dokumenty

Dokumenty

Projekt:  Kategorie:

Soubor:

Projekt:	Kategorie:	Dokument:	Datum:	Vložil:
Projekt 1x	Kategorie 1	<a href="#">cabel_pvps.odt</a>	2010-04-03	Jakub Měchura 

Obrázek 63- IS projekty - sekce dokumenty

1. Jestliže má uživatel právo vkládat, je mu zobrazen formulář pro vkládání dokumentu.
2. Uživatel vybere projekt, kategorii v projektu, nahraje soubor a stiskne tlačítko „Vytvořit“. Výběr kategorie není povinný. Jestliže nebude vybrána kategorie, bude dokument automaticky vložen do kategorie „Dokumenty“, která je automaticky vytvořena pro všechny projekty.
3. Dokumenty lze setřídovat abecedně podle sloupců v seznamu „vložených dokumentů“.
4. Uživatel s právem mazání má možnost mazat dokumenty.

### 3.5.1.7 IS – sekce Pošta

[Poslat novou](#)   [Nepřečtená\(0\)](#)   [Přečtená](#)   [Odeslaná](#)



Komu:  Všem lidem z projektu:

Příloha:

Předmět:

Obrázek 64 - IS projekty - sekce pošta

1. Tato sekce se skládá ze 4. částí a to: Poslat novou, Nepřečtená, Přečtená, Odeslaná.
2. Uživatel má možnost poslat poštu včetně přílohy a to buď:
  - a) Jednomu uživateli
  - b) Více uživatelům odděleným středníkem
  - c) Všem lidem z projektu pomocí výběru ze seznamu „Všem lidem z projektu“.
3. Sekce nepřečtená pošta obsahuje všechnu nepřečtenou poštu uživatele.
4. Sekce přečtená pošta obsahuje všechnu přečtenou poštu uživatele.
5. Sekce odeslaná pošta obsahuje všechnu odeslanou poštu uživatele.

<a href="#">Poslat novou</a>	<a href="#">Nepřečtená(0)</a>	<a href="#">Přečtená</a>	<a href="#">Odeslaná</a>	
Datum:	Adresát:	Předmět:	Stav:	Akce:
2010-04-08	mol049	zkouska4	neprecteno	 

Obrázek 65 - IS projekty - sekce nepřečtená pošta

Odesílatel:	mec063
Adresát:	mol049
Předmět:	zkouska4
Příloha:	
sc	

Obrázek 66 - IS projekty - detail pošty

### 3.5.1.8 IS – sekce Osobní

Login:	<input type="text" value="mec063"/>
Jméno:	<input type="text" value="Jakub"/>
Příjmení:	<input type="text" value="Měchura"/>
Telefon:	<input type="text"/>
Email:	<input type="text" value="mec063@vsb.cz"/>
	<input type="button" value="Změnit"/>

Obrázek 67 - IS projekty - sekce osobní

1. Zde má uživatel možnost změnit údaje o sobě, a to telefon a email.

### 3.5.2 Programátorská příručka

Programátorská příručka je umístěna na přiloženém CD ve složce **/Projekt2/Programátorská příručka**

### 3.5.3 Popis instalace

Z důvodu toho, že je v rámci tohoto IS využito protokolu LDAP, nelze tento IS odzkoušet tak jako předchozí příklad pomocí programu VertrigoServ na lokálním serveru. Pro odzkoušení tohoto projektu je potřeba nasadit IS ostrého provozu. K tomu je potřeba:

- 1) Umístit na daný webový server předchozí IS, postup je stejný.
  - a) V databázi spustit skript **skript.sql** umístěný na přiloženém CD ve složce **/Projekt1/sql**
  - b) Do adresáře na webovém serveru zkopírovat složku **/bak** umístěnou na CD ve složce **/Projekt1/**
- 2) Spustit skript, který je umístěný na přiloženém CD ve složce Projekt2/sql/skript.sql
- 3) Do předchozího IS, složky **/bak** zkopírovat složku **projekty**, která se nachází na přiloženém **/Projekt2/**

Výsledný IS se nachází na <http://homel.vsb.cz/~mec063/bak/projekty>

Pro odzkoušení systému je zde vytvořen testovací login user01 s heslem user01.





## 4 Závěr

Cílem této bakalářské práce bylo provést modernizaci webových stránek, jejíž hlavní a podstatnou částí bylo vytvořit dynamicky generované stránky a dva informační systémy. První informační systém evidence kurzů a podávání přihlášek na ty to kurzy byl spjat s přechodem statických stránek na dynamicky generované, kde správcem vytvářené kategorie, kurzy a termíny jsou automaticky generovány na webových stránkách.

Druhý IS je opět umístěn na webových stránkách a slouží k evidenci projektů.

Důležitou částí vývoje IS pak byly analýzy. Datová, zabývající se analýzou databáze na logické úrovni a funkční, která se týká popisu procesů a toku dat v IS. Bez těchto analýz by nebylo možno implementovat IS.

Na základě takto vytvořených analýz jsem následně provedl bez předchozích znalostí tvorby IS implementaci obou IS. Také implementace pomocí jazyka PHP pro mě byla novinkou. K nabytí znalosti nutných pro tvorbu IS jsem využil informace z literatury a webových stránek, které lze najít v kapitole **5 Zdroje a přílohy**.

V budoucnosti se na webových stránkách chystá také implementace RSS čtečky a kalendáře akcí.



## 5 Zdroje a přílohy

### 5.1 Literatura

- [1] ELIZABETH NARAMORE, Timothy Boronczyk, et al. *PHP 6, MySQL, Apache : Vytváříme webové aplikace*. Brno : Computer press, a.s., 2009. 816 s. ISBN 978-80-251-2767-4.
- [2] KOFLER, Michael; ÖGGL, Bernd. *PHP5 a MySQL 5 : Průvodce webového programátora*. Brno : Computer press, a.s., 2007. 607 s. ISBN 978-80-251-1813-9.

### 5.2 Internet

- [3] BOUŠKA, Petr. *SAMURAJ-cz.com : počítačové sítě, Cisco, administrace, webcoding a karate* [online]. 2006 [cit. 2007-11-02]. Jak na ldap a ldaps v php pod windows. Dostupné z WWW: <<http://www.samuraj-cz.com/clanek/jak-na-ldap-a-ldaps-v-php-pod-windows/>>.
- [4] LDAP In *Wikipedia : the free encyclopedia* [online]. St. Petersburg (Florida) : Wikipedia Foundation, 7. 5. 2007, 26. 2. 2010 [cit. 2010-04-05]. Dostupné z WWW: <<http://cs.wikipedia.org/wiki/LDAP>>.
- [5] Php In *Wikipedia : the free encyclopedia* [online]. St. Petersburg (Florida) : Wikipedia Foundation, 2. 6. 2004, 24. 3. 2010 [cit. 2010-04-05]. Dostupné z WWW: <<http://cs.wikipedia.org/wiki/Php>>.
- [6] Mysql In *Wikipedia : the free encyclopedia* [online]. St. Petersburg (Florida) : Wikipedia Foundation, 21. 11. 2004, 4. 4. 2010 [cit. 2010-04-05]. Dostupné z WWW: <<http://cs.wikipedia.org/wiki/Mysql>>.
- [7] Javascript In *Wikipedia : the free encyclopedia* [online]. St. Petersburg (Florida) : Wikipedia Foundation, 12. 8. 2004, 1. 3. 2010 [cit. 2010-04-05]. Dostupné z WWW: <<http://cs.wikipedia.org/wiki/Javascript>>.
- [8] Xhtml In *Wikipedia : the free encyclopedia* [online]. St. Petersburg (Florida) : Wikipedia Foundation, 31. 10. 2004, 1. 4. 2010 [cit. 2010-04-05]. Dostupné z WWW: <<http://cs.wikipedia.org/wiki/Xhtml>>.
- [9] *PHP : Hypertext Preprocessor* [online]. 2001 [cit. 2010-04-05]. [Http://cz.php.net/](http://cz.php.net/). Dostupné z WWW: <<http://cz.php.net/>>.
- [10] ŠARMANOVÁ, Jana. *Databázové a informační systémy* [online]. první. VŠB – Technická univerzita Ostrava: [s.n.], 2007 [cit. 2010-04-05]. Dostupné z WWW: <<http://www.elearn.vsb.cz/archivcd/FEI/DAIS/>>. ISBN 978-80-248-1499-5.
- [11] ŠARMANOVÁ, Jana. *Teorie zpracování dat* [online]. první. VŠB – Technická univerzita Ostrava: VŠB – Technická univerzita Ostrava: [s.n.], 2003 [cit. 2010-04-05]. Dostupné z WWW: <<http://www.elearn.vsb.cz/archivcd/FEI/TZD/>>.
- [12] JANOVSKEÝ, Dušan. *Jak psát web : návod na html stránky* [online]. 2003 [cit. 2010-04-05]. [Http://www.jakpsatweb.cz/](http://www.jakpsatweb.cz/). Dostupné z WWW: <<http://www.jakpsatweb.cz/>>. ISSN 1801-0458.

- [13] *JakNaWeb.com : Jak na web stránky a tvorba www stránek pomocí XHTML, CSS, PHP, MySQL, .NET, SEO a grafika* [online]. 2002 [cit. 2010-04-05]. [Http://www.jaknaweb.com/](http://www.jaknaweb.com/). Dostupné z WWW: [<http://www.jaknaweb.com/>](http://www.jaknaweb.com/).
- [14] *MySQL :: The world's most popular open source database* [online]. 2008 [cit. 2010-04-24]. [Http://mysql.com/](http://mysql.com/). Dostupné z WWW: [<http://dev.mysql.com/doc/refman/5.0/en/index.html>](http://dev.mysql.com/doc/refman/5.0/en/index.html).
- [15] BOUŠKA, Petr. *SAMURAJ-cz.com : počítačové sítě, Cisco, administrace, webcoding a karate* [online]. 2006 [cit. 2007-11-02]. Adresářové služby a LDAP. Dostupné z WWW: [<http://www.samuraj-cz.com/clanek/adresarove-sluzby-a-ldap/>](http://www.samuraj-cz.com/clanek/adresarove-sluzby-a-ldap/).

## 5.3 Seznam příloh na CD

**bc.doc** – text bakalářské práce (word 97 – 03)

**bc.docx** - text bakalářské práce

**zadání.pdf** – zadání bakalářské práce

**/Software/** – software VertrigoServ

**/Projekt1/sql/** - složka obsahuje skript pro vytvoření databáze IS evidence kurzů a přihlášek na kurzy

**/Projekt1/programátorská příručka/** - programátorská příručka k IS evidence kurzů a přihlášek na kurzy

**/Projekt1/bak/** - zdrojové kódy pro IS evidence kurzů a přihlášek na kurzy

**/Projekt2/sql/** - složka obsahuje skript pro vytvoření databáze IS evidence projektů

**/Projekt2/programátorská příručka/** - programátorská příručka k IS evidence projektů

**/Projekt2/projekty/** - zdrojové kódy pro IS evidence projektů